

27123

PATENT TRADEMARK OFFICE

Docket No. 1232-4658

2622

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant(s): FUKASAWA, et al.

Group Art Unit: **RECEIVED**

Serial No.: 09/708,907

Examiner: JUN 15 2001

Filed: November 8, 2000

Technology Center 2600

For: IMAGE PROCESSING SYSTEM, IMAGE PROCESSING METHOD, IMAGE  
UPLOAD SYSTEM, STORAGE MEDIUM, AND IMAGE UPLOAD SERVER**CERTIFICATE OF MAILING (37 C.F.R. §1.8(a))**Commissioner for Patents  
Washington, D.C. 20231**RECEIVED**

JUN 28 2001

Sir:

Technology Center 2600

I hereby certify that the attached:

1. Claim to Convention Priority w/three documents
2. Return Postcard Receipt
- 3.

along with any paper(s) referred to as being attached or enclosed and this Certificate of Mailing are being deposited with the United States Postal Service on date shown below with sufficient postage as first-class mail in an envelope addressed to the: Commissioner for Patents, Washington, D.C., 20231.

Respectfully submitted,  
MORGAN & FINNEGAN, L.L.P.

Dated: June 6, 2001

By:

  
Helen TigerCorrespondence Address:MORGAN & FINNEGAN, L.L.P.  
345 Park Avenue  
New York, NY 10154-0053  
(212) 758-4800 Telephone  
(212) 751-6849 Facsimile



27123

PATENT TRADEMARK OFFICE

RECEIVED

Docket No. 1232-46587001  
JUN 13 2001

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant(s): FUKASAWA, et al.

Group Art Unit: 2622

Serial No.: 09/708,907

Examiner:

Filed: November 8, 2000

For: IMAGE PROCESSING SYSTEM, IMAGE PROCESSING METHOD, IMAGE  
UPLOAD SYSTEM, STORAGE MEDIUM, AND IMAGE UPLOAD SERVER

**CLAIM TO CONVENTION PRIORITY**

Commissioner for Patents  
Washington, D.C. 20231

RECEIVED

JUN 28 2001

Technology Center 2600

Sir:

In the matter of the above-identified application and under the provisions of 35 U.S.C. §119 and 37 C.F.R. §1.55, applicant(s) claim(s) the benefit of the following prior application(s):

Application(s) filed in: JAPAN  
In the name of: Canon Kabushiki Kaisha  
Serial No(s): 11-323850  
11-325339  
2000-316647

Filing Date(s): November 15, 1999  
November 16, 1999  
October 17, 2000

☒ Pursuant to the Claim to Priority, applicant(s) submit(s) a duly certified copy of said foreign application.

☐ A duly certified copy of said foreign application is in the file of application  
Serial No. \_\_\_\_\_, filed \_\_\_\_\_.

Respectfully submitted,  
MORGAN & FINNEGAN, L.L.P.

Dated: June 6, 2001

By: \_\_\_\_\_

  
Joseph A. Calvaruso  
Registration No. 28,287

Correspondence Address:

MORGAN & FINNEGAN, L.L.P.  
345 Park Avenue  
New York, NY 10154-0053  
(212) 758-4800 Telephone  
(212) 751-6849 Facsimile

CF014923 US



日 本 国 特 許 庁

PATENT OFFICE  
JAPANESE GOVERNMENT

1232-4658

別紙添付の書類に記載されている事項は下記の出願書類に記載されて  
いる事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed  
with this Office.

出 願 年 月 日  
Date of Application:

1999年11月15日

出 願 番 号  
Application Number:

平成11年特許願第323850号

出 願 人  
Applicant(s):

キヤノン株式会社

RECEIVED

JUN 28

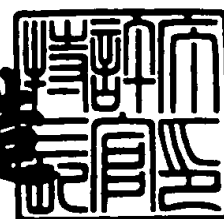
Technology Center 2600

CERTIFIED COPY OF  
PRIORITY DOCUMENT

2000年12月 8日

特 許 庁 長 官  
Commissioner,  
Patent Office

及 川 耕 造



【書類名】 特許願

【整理番号】 3937010

【提出日】 平成11年11月15日

【あて先】 特許庁長官 近藤 隆彦 殿

【国際特許分類】 H04N 1/00

【発明の名称】 映像処理システム、映像処理方法、映像アップロードシステム、記憶媒体、映像アップロードサーバ

【請求項の数】 42

【発明者】

【住所又は居所】 東京都大田区下丸子3丁目30番2号キヤノン株式会社  
内

【氏名】 深澤 寿彦

【特許出願人】

【識別番号】 000001007

【住所又は居所】 東京都大田区下丸子3丁目30番2号

【氏名又は名称】 キヤノン株式会社

【代表者】 御手洗 富士夫

【電話番号】 03-3758-2111

【代理人】

【識別番号】 100090538

【住所又は居所】 東京都大田区下丸子3丁目30番2号キヤノン株式会社  
内

【弁理士】

【氏名又は名称】 西山 恵三

【電話番号】 03-3758-2111

【選任した代理人】

【識別番号】 100096965

【住所又は居所】 東京都大田区下丸子3丁目30番2号キヤノン株式会  
社内

【弁理士】

【氏名又は名称】 内尾 裕一

【電話番号】 03-3758-2111

【選任した代理人】

【識別番号】 100110009

【住所又は居所】 東京都大田区下丸子3丁目30番2号キャノン株式会社  
社内

【弁理士】

【氏名又は名称】 青木 康

【電話番号】 03-3758-2111

【選任した代理人】

【識別番号】 100069877

【住所又は居所】 東京都大田区下丸子3丁目30番2号キャノン株式会社  
社内

【弁理士】

【氏名又は名称】 丸島 儀一

【電話番号】 03-3758-2111

【手数料の表示】

【予納台帳番号】 011224

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 9908388

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 映像処理システム、映像処理方法、映像アップロードシステム、記憶媒体、映像アップロードサーバ

【特許請求の範囲】

【請求項 1】 ネットワーク上に配置された映像入力装置から、映像データを取得する取得手段と、

前記取得手段によって得られた複数の前記映像データと、前記映像データとは別に当該映像取得時の情報を格納する格納手段と、

前記取得手段により得られた前記映像データと、当該映像取得時の情報とから、表示用映像制御データを作成する作成手段と、

前記作成手段によって作成された表示用映像制御データを送信する送信手段と

を有することを特徴とする映像処理システム。

【請求項 2】 さらに、予め決められたスケジュールに従って、前記取得手段を制御する映像取得制御手段を有することを特徴とする請求項 1 に記載の映像処理システム。

【請求項 3】 前記作成手段は、前記映像データ取得時の取得日時、映像データのサイズ、タイトル、カメラ制御情報のうち、少なくともひとつの情報を、前記映像データとともに表示させるための所定のテンプレートに従って、前記表示用映像制御データを作成することを特徴とする請求項 1 乃至 2 に記載の映像処理システム。

【請求項 4】 さらに、前記取得手段によって取得された映像データを加工編集して、インターネットへ向けて配信するのに適した形式のデータを生成する生成手段を有し、前記格納手段は、前記生成手段によって生成されたデータを格納することを特徴とする請求項 1 乃至 3 のいずれかに記載の映像処理システム。

【請求項 5】 さらに、前記ネットワークからの表示用映像の送信要求を受け付けて、当該送信要求先へ、当該表示用映像制御データを送信すべく前記送信手段を制御する手段を有することを特徴とする請求項 1 乃至 4 のいずれかに記載の映像処理システム。

【請求項 6】 さらに、送信先までのデータ送信経路を管理する送信経路管理手段を有することを特徴とする請求項 1 乃至 5 のいずれかに記載の映像処理システム。

【請求項 7】 前記送信経路管理手段は、少なくとも、モデム接続のための送信経路と、ゲートウェイ・ホスト接続のための送信経路を管理することを特徴とする請求項 6 に記載の映像処理システム。

【請求項 8】 さらに、ネットワーク上のユーザーからの指示を受け付けて、前記取得手段を制御して、前記スケジュールとは関係なく、任意のタイミングで映像データを取得させる手段を有することを特徴とする請求項 2 乃至 7 のいずれかに記載の映像処理システム。

【請求項 9】 前記格納手段は、前記取得手段から、ネットワークを介して前記映像データを受け取ることを特徴とする請求項 1 乃至 8 のいずれかに記載の映像処理システム。

【請求項 10】 さらに、前記スケジュールに従って、所定のサーバーへ、前記映像データを送信するよう制御する、送信制御手段を有することを特徴とする請求項 1 乃至 9 のいずれかに記載の映像処理システム。

【請求項 11】 さらに、前記映像データを加工編集するアプリケーションプログラムを制御する情報を含むことを特徴とする請求項 1 乃至 10 のいずれかに記載の映像処理システム。

【請求項 12】 前記取得手段と、前記格納手段と、前記作成手段と、前記送信手段とで実行される処理を、ネットワーク上に別個に配置された複数の映像処理装置で実行させることを特徴とする請求項 4 に記載の映像処理システム。

【請求項 13】 前記作成手段は、HTML ファイルを作成することを特徴とする請求項 1 乃至 12 のいずれかに記載の映像処理システム。

【請求項 14】 前記作成手段は、XML ファイルを作成することを特徴とする請求項 1 乃至 13 のいずれかに記載の映像処理システム。

【請求項 15】 さらに、前記格納手段に格納された映像データの有効期限を管理し、前記有効期限の経過した映像を、前記格納手段から消去する消去手段を有することを特徴とする請求項 1 乃至 14 のいずれかに記載の映像処理システム。



ム。

【請求項 1 6】 ネットワーク上に配置された映像入力装置から、映像データを取得する取得手段と、

前記取得手段によって得られた複数の前記映像データを格納する格納手段と、  
予め指定されたスケジュールに従って、前記取得手段を制御する映像取得制御手段と、

前記ネットワークから、前記格納手段に格納された前記映像データを所定の条件に従って送信する送信手段と、

前記格納手段に格納された映像データの期限を管理する手段とを有することを特徴とする映像処理システム。

【請求項 1 7】 ネットワーク上に配置された映像入力装置から、映像データを取得する取得手段と、

前記取得手段によって得られた複数の前記映像データを格納する格納手段と、  
予め指定されたスケジュールに従って、前記取得手段を制御する映像取得制御手段と、

前記ネットワークから、前記格納手段に格納された前記映像データの加工編集要求を受け付けて、当該映像データを加工編集する加工編集手段と、

前記加工編集手段によって加工編集された前記映像データを送信する送信手段と、

を有することを特徴とする映像処理システム。

【請求項 1 8】 ネットワーク上に配置された映像入力装置から、映像データを取得する取得手段と、

前記取得手段によって得られた複数の前記映像データを格納する格納手段と、  
予め指定されたスケジュールに従って、前記取得手段を制御する映像取得制御手段と、

前記取得手段により得られた前記映像データと、当該映像取得時の情報とから、表示用映像制御データを作成する作成手段と、

前記作成手段によって作成された表示用映像制御データを送信先へ向けて送信する送信手段と、

前記送信手段による送信処理の送信先及び、送信時間を制御する送信制御手段とを有することを特徴とする映像処理システム。

【請求項 1 9】 ネットワーク上に配置された映像入力装置から、映像データを取得する取得手段と、

前記取得手段によって得られた複数の前記映像データを格納する格納手段と、  
予め指定されたスケジュールに従って、前記取得手段を制御する映像取得制御手段と、

前記取得手段により得られた前記映像データと、当該映像取得時の情報とから、表示用映像制御データを作成する作成手段と、

前記作成手段によって作成された表示用映像制御データを送信する送信手段と

を有することを特徴とする映像アップロードサーバ。

【請求項 2 0】 ネットワーク上に配置された映像入力装置から、映像データを取得する取得工程と、

前記取得工程で得られた複数の前記映像データと、前記映像データとは別に当該映像取得時の情報を格納する格納工程と、

前記取得工程で得られた前記映像データデータと、当該映像取得時の情報とから、表示用映像制御データを作成する作成工程と、

前記作成工程で作成された表示用映像制御データを送信する送信工程と、  
を有することを特徴とする映像処理方法。

【請求項 2 1】 さらに、予め決められたスケジュールに従って、前記取得工程を制御する映像取得制御工程を有することを特徴とする請求項 2 0 に記載の映像処理方法。

【請求項 2 2】 前記作成工程は、前記映像データ取得時の取得日時、映像データのサイズ、タイトル、カメラ制御情報のうち、少なくともひとつの情報を、前記映像データとともに表示させるための所定のテンプレートに従って、前記表示用映像制御データを作成することを特徴とする請求項 2 0 乃至 2 1 に記載の映像処理方法。

【請求項 2 3】 さらに、前記取得工程によって取得された映像データを加

工編集して、インターネットへ向けて配信するのに適した形式のデータを生成する生成工程を有し、前記格納工程は、前記生成工程によって生成されたデータを格納することを特徴とする請求項 2 0 乃至 2 2 のいずれかに記載の映像処理方法。

【請求項 2 4】 さらに、前記ネットワークからの表示用映像の送信要求を受け付けて、当該送信要求先へ、当該表示用映像制御データを送信すべく前記送信工程を制御する工程を有することを特徴とする請求項 2 0 乃至 2 3 のいずれかに記載の映像処理方法。

【請求項 2 5】 さらに、送信先までのデータ送信経路を管理する送信経路管理工程を有することを特徴とする請求項 2 0 乃至 2 4 のいずれかに記載の映像処理方法。

【請求項 2 6】 前記送信経路管理工程は、少なくとも、モデム接続のための送信経路と、ゲートウェイ・ホスト接続のための送信経路を管理することを特徴とする請求項 2 5 に記載の映像処理方法。

【請求項 2 7】 さらに、ネットワーク上のユーザーからの指示を受け付けて、前記取得工程を制御して、前記スケジュールとは関係なく、任意のタイミングで映像データを取得させる工程を有することを特徴とする請求項 2 1 乃至 2 6 のいずれかに記載の映像処理方法。

【請求項 2 8】 前記格納工程は、前記取得工程から、ネットワークを介して前記映像データを受け取ることを特徴とする請求項 2 0 乃至 2 7 のいずれかに記載の映像処理方法。

【請求項 2 9】 さらに、前記スケジュールに従って、所定のサーバーへ、前記映像データを送信するよう制御する、送信制御工程を有することを特徴とする請求項 2 0 乃至 2 8 のいずれかに記載の映像処理方法。

【請求項 3 0】 さらに、前記映像データを加工編集するアプリケーションプログラムを制御する情報を含むことを特徴とする請求項 2 0 乃至 2 9 のいずれかに記載の映像処理方法。

【請求項 3 1】 前記取得工程と、前記格納工程と、前記作成工程と、前記送信工程とで実行される処理を、ネットワーク上に別個に配置された複数の映像

処理装置で実行させることを特徴とする請求項 2 3 に記載の映像処理方法。

【請求項 3 2】 前記作成工程は、HTML ファイルを作成することを特徴とする請求項 2 0 乃至 3 1 のいずれかに記載の映像処理方法。

【請求項 3 3】 前記作成工程は、XML ファイルを作成することを特徴とする請求項 2 0 乃至 3 2 のいずれかに記載の映像処理方法。

【請求項 3 4】 さらに、前記格納工程に格納された映像データの有効期限を管理し、前記有効期限の経過した映像を、前記格納工程から消去する消去工程を有することを特徴とする請求項 2 0 乃至 3 3 のいずれかに記載の映像処理方法。

【請求項 3 5】 ネットワーク上に配置された映像入力装置から、映像データを取得する取得工程と、

前記取得工程で得られた複数の前記映像データを格納する格納工程と、

予め指定されたスケジュールに従って、前記取得工程を制御する映像取得制御工程と、

前記ネットワークから、前記格納工程で格納された前記映像データを所定の条件に従って送信する送信工程と、

前記格納工程で格納された映像データの期限を管理する工程とを有することを特徴とする映像処理方法。

【請求項 3 6】 ネットワーク上に配置された映像入力装置から、映像データを取得する取得工程と、

前記取得工程によって得られた複数の前記映像データを格納する格納工程と、

予め指定されたスケジュールに従って、前記取得工程を制御する映像取得制御工程と、

前記ネットワークから、前記格納工程に格納された前記映像データの加工編集要求を受け付けて、当該映像データを加工編集する加工編集工程と、

前記加工編集工程によって加工編集された前記映像データを送信する送信工程と、

を有することを特徴とする映像処理方法。

【請求項 3 7】 ネットワーク上に配置された映像入力装置から、映像デー

タを取得する取得工程と、

前記取得工程によって得られた複数の前記映像データを格納する格納工程と、

予め指定されたスケジュールに従って、前記取得工程を制御する映像取得制御工程と、

前記取得工程により得られた前記映像データに、当該映像取得時の情報を付加して、表示用映像制御データを作成する作成工程と、

前記作成工程によって作成された表示用映像制御データを送信先へ向けて送信する送信工程と、

前記送信工程による送信処理の送信先及び、送信時間を制御する送信制御工程とを有することを特徴とする映像処理方法。

【請求項 38】 コンピュータが読み込み実行することで、映像処理方法を実現させるプログラムが記録された記録媒体であって、

ネットワーク上に配置された映像入力装置から、映像データを取得する取得工程と、

前記取得工程で得られた複数の前記映像データと、前記映像データとは別に当該映像取得時の情報を格納する格納工程と、

前記取得工程で得られた前記映像データデータと、当該映像取得時の情報とから、表示用映像制御データを作成する作成工程と、

前記作成工程で作成された表示用映像制御データを送信する送信工程と、を有することを特徴とする映像処理方法を実現させるプログラムが記録された記録媒体。

【請求項 39】 コンピュータが読み込み実行することで、映像処理方法を実現させるプログラムが記録された記録媒体であって、

ネットワーク上に配置された映像入力装置から、映像データを取得する取得工程と、

前記取得工程で得られた複数の前記映像データを格納する格納工程と、

予め指定されたスケジュールに従って、前記取得工程を制御する映像取得制御工程と、

前記ネットワークから、前記格納工程で格納された前記映像データを所定の条

件に従って送信する送信工程と、

前記格納工程で格納された映像データの期限を管理する工程とを有することを特徴とする映像処理方法を実現させるプログラムが記録された記録媒体。

【請求項 4 0】 コンピュータが読み込み実行することで、映像処理方法を実現させるプログラムが記録された記録媒体であって、

ネットワーク上に配置された映像入力装置から、映像データを取得する取得工程と、

前記取得工程によって得られた複数の前記映像データを格納する格納工程と、

予め指定されたスケジュールに従って、前記取得工程を制御する映像取得制御工程と、

前記ネットワークから、前記格納工程に格納された前記映像データの加工編集要求を受け付けて、当該映像データを加工編集する加工編集工程と、

前記加工編集工程によって加工編集された前記映像データを送信する送信工程と、

を有することを特徴とする映像処理方法を実現させるプログラムが記録された記録媒体。

【請求項 4 1】 コンピュータが読み込み実行することで、映像処理方法を実現させるプログラムが記録された記録媒体であって、

ネットワーク上に配置された映像入力装置から、映像データを取得する取得工程と、

前記取得工程によって得られた複数の前記映像データを格納する格納工程と、

予め指定されたスケジュールに従って、前記取得工程を制御する映像取得制御工程と、

前記取得工程により得られた前記映像データに、当該映像取得時の情報を付加して、表示用映像制御データを作成する作成工程と、

前記作成工程によって作成された表示用映像制御データを送信先へ向けて送信する送信工程と、

前記送信工程による送信処理の送信先及び、送信時間を制御する送信制御工程とを有することを特徴とする映像処理方法を実現させるプログラムが記録された記

録媒体。

【請求項 4 2】 ネットワーク上に配置された映像入力装置から、複数の前記映像データと、当該映像取得時の情報を取得する取得手段と、

前記取得手段により得られた前記映像データと、当該映像取得時の情報とから、表示用映像制御データを作成する作成手段と、

前記作成手段によって作成された前記表示用制御データを格納する格納手段とを有することを特徴とする映像アップロードシステム。

【発明の詳細な説明】

【0 0 0 1】

【発明の属する技術分野】

本発明は、コンピュータネットワークにおける映像発信サービスに関するものである。特に、定期的にビデオカメラ等の映像を取得し、不特定ユーザにその映像データを提供するサービスに関するものである。

【0 0 0 2】

【従来の技術】

インターネット、イントラネットなどのコンピュータネットワークは、WWW（ワールドワイドウェブ）の出現によって飛躍的に普及した。WWWにより情報を不特定多数の人間に提供し、また不特定多数の人間から必要な情報を取得することが可能になった。

【0 0 0 3】

通常、WWWはあらかじめ用意されているテキスト、画像、音声を不特定多数のユーザに提供することができる。WWWの応用形態の多様化は、あらかじめ用意されていたデータではなく、リアルタイムに映像、音声等のデータを取得し、これを公開したいという要求を生み出した。

【0 0 0 4】

この要求に対する1つの解答として映像アップロードサービスが考え出され製品化されてきた。この映像アップロードサービスは、ビデオカメラから定期的に映像データを取得し、この映像データをWWWサーバに送信(アップロード)するものである。

【 0 0 0 5 】

図4は一般的な映像アップロードサービスの概要をあらわす図である。

【 0 0 0 6 】

映像アップロードホスト(401)は映像アップロードのサービスを提供する機能をもったコンピュータである。映像アップロードホスト(401)にはビデオカメラ(402)が接続されている。映像アップロードホスト(401)はビデオカメラ(402)から、所定のスケジュールに従って画像を取得(キャプチャ)する。キャプチャされた画像データは、コンピュータネットワーク(404)を経由してWWWホスト(403)に送信される。(この一連の処理のことを「映像アップロード処理」とよぶ。)

【 0 0 0 7 】

WWWホスト(403)は、送信されてきた映像データをハードディスク等に記録する。映像アップロードサービスの利用者(406)はInternet Explorer(TM)等のWebブラウザ(405)を利用することで、WWWホスト(403)に蓄積されている映像データを見ることができる。

【 0 0 0 8 】

図5は、Webブラウザでの映像データの表示例である。映像データを表示するためのウィンドウ(501)が1つオープンされている。ウィンドウ(501)上には、映像データ(502)と、映像に関連している文字情報(503、504)が表示される。

【 0 0 0 9 】

506は、ウィンドウ(501)でどのようにデータを表示すべきかを定義したHTMLファイルの内容である。506内の行(507)はウィンドウ上の文字情報(503)の表示内容と表示方式を指定している。行(509)は文字情報(504)の表示内容と表示方式を指定している。画像データ(502)の定義は行(508)でおこなわれている。

【 0 0 1 0 】

【発明が解決しようとする課題】

しかし、従来のアップロードシステムでは、ユーザーが最新の画像を見ることが出来るが、以前撮影された画像を見ることが出来なかった。

【 0 0 1 1 】

また、アップロードされた映像データは、何時、何処でどのようにして取得さ



れた(とられた)ものかを表示したいという要望がある。また、過去の映像データを一覧表形式などで表示したいという要求も存在する。

【0 0 1 2】

しかし、映像データを表示するための506のようなHTMLファイルは、WWWホスト(403)上にあらかじめ作成しておかなければならなかった。このことは、(503)や(504)のようにあらかじめ用意しておくことのできる情報(例えば、撮影位置などの映像によって変わらない情報)を提供する上では問題はない。

【0 0 1 3】

しかし、映像データがWWWホスト(403)に記録されてからでないと値が分からない情報もある。例えば映像データの取得時刻や映像データのサイズなどがこのような情報である。これらの情報をあらかじめ用意したHTMLファイル内に埋めこむことはできないので、このような情報を表示するためには、映像を撮り込んだ後に、管理者がわざわざ、HTML文書を更新しなければならなかった。又、撮影日程度は、カメラでの写し込みによって提供することは、可能であるが、この場合は、映像に文字が必ず、入ってしまい、純粹に映像だけを提供することはできない。

【0 0 1 4】

つまり、従来のアップロードサービスは、映像データのファイルをアップロードするだけであり、それを表示するページはWWWサーバにあらかじめ用意する必要があった。この方式の欠点は、前もってページを作成しておかなければならなため、映像データを取得してからでなければ分からない情報 --- 例えば、映像データの取得日時、データサイズ ---などの情報を表示(提供)することができないことである。

【0 0 1 5】

また、表示できる映像ファイルの個数も固定されることになり、任意の枚数の画像を、一覧形式で見ることが出来ないし、新しい映像ファイルがアップロードされるごとに表示画像の個数を増やすといったことも実現できない。

【0 0 1 6】

さらに、取得した映像データに対して、日付やメッセージを埋め込んだり、複

数の映像データをつなぎあわせてパノラマ画像を作成したいといった、映像データの加工要求もある。

【0017】

そこで、本発明は、以前に撮影された映像も閲覧可能にした、新たなサービスを提供するアップロードシステムを提供し、また、映像撮影時の情報も容易に提供でき、多様な表示形式や加工要求に応えることが出来るアップロードシステムを提供することを目的とする。

【0018】

また、過去に撮影した映像を閲覧可能にしたとしても、例えば定期的にファイルをアップロードすることにより、映像ファイルがWWWホストのハードディスクに蓄積されつづけ、ついにはディスク容量オーバーしてしまい、新たな映像を取込めなくなることを防止することを、本発明の他の目的とする。

【0019】

さらに、WWWサービスを享受するユーザが多くなったり、ユーザーに提供するデータ量の増加や、提供サービスの形態の増加につれて、単独のWWWサーバでは、十分な対応が出来なくなることも予想される。

【0020】

しかし、従来のアップロードサービスは、WWWサーバとアップロードサービスが同一ホストで動作することを前提としていたり、そうでなくても、アップロード先が1つに固定されていたりしているので、複雑なWWWサーバの運用形態に対応することはできない。

【0021】

本発明は、複数のWWWサーバで運用可能なアップロードサービスを提供することを、更に他の目的とする。

【0022】

【課題を解決するための手段】

上記の課題を解決する為、本発明の映像処理システムは、ネットワーク上に配置された映像入力装置から、映像データを取得する取得手段と、取得手段によって得られた複数の映像データと、映像データとは別に当該映像取得時の情報を格

納する格納手段と、取得手段により得られた映像データと、当該映像取得時の情報とから、表示用映像制御データを作成する作成手段と、作成手段によって作成された表示用映像制御データを送信する送信手段とを有することを特徴とする。

【 0 0 2 3 】

【発明の実施の形態】

〔第1実施例〕

ここでは、第1の実施例としての映像アップロード・サーバについて説明する。まず、以降の説明で使用する用語について簡単に説明する。

【 0 0 2 4 】

－ 映像アップロード

映像データをビデオカメラ等から取得し、それをWWWホストなど利用者にとって都合の良いホストに送信すること。受け取ったホストは映像データを2次記憶装置に蓄積するなどの処理を行なう。

【 0 0 2 5 】

－ 時点

日付と時刻で指定できるある時間軸上の1点の意である。曜日を含めることもある。通常は、「日付」や「時刻」という言葉は、それ自体である時点をあらわしているときもあれば、時点の構成要素としての日付と時刻をあらわすこともある。混乱をさけるために、以降で「日付」、「時刻」と表記する場合は、かならず「時点」の構成要素としての日付、時点をあらわすものとする。

【 0 0 2 6 】

－ オブジェクト管理

あるオブジェクトのグループを管理することである。すなわち、必要に応じて「グループにオブジェクトを追加する」、「グループからオブジェクトを削除する」、「グループからある条件を満たすオブジェクトを取り出す」、「グループ内のオブジェクトに適切な処理を適用する。(属性値を書き換えるなど)」の処理をおこなう。

【 0 0 2 7 】

実装においては、グループに属するオブジェクトのオブジェクトIDをリスト

やテーブルなどで保持してやれば簡単に実現することができる。「管理モジュール」いう名前が付けられているモジュールは必ずこの機能を持っている。

【0028】

- オブジェクト/オブジェクトID/オブジェクト領域

データをメインメモリにおいた時の形態である。「オブジェクトID」で参照する。オブジェクトの置かれているメモリ上の領域のことを「オブジェクト領域」と呼ぶ。多くの場合、オブジェクトが置かれているメモリ領域の先頭アドレスをオブジェクトIDとして利用する。

【0029】

- オブジェクトの属性/属性値

オブジェクトを構成しているデータのそれぞれに言及するとき、そのデータのことを「オブジェクトの属性」あるいは単に「属性」と呼ぶ。例えば、「ファイルオブジェクト」は「属性」として「ファイル名」を持っている。」といった使い方をする。属性の具体的な値について述べる時は、その値のことを「属性値」と呼ぶ。(実装上は、オブジェクト指向言語におけるインスタンス変数とその値に相当する。)

【0030】 - ソフトウェア・モジュール/サブモジュール

単に「モジュール」とも呼ぶ。ソフトウェアの論理的な構成要素である。プログラムの実装における「(1つの)関数」、「複数の関数あつまり(関数群)」、「ソースファイル」、「オブジェクト指向言語のクラス」などに対応する。

【0031】

モジュールはさらに小さな構成要素から構成されることもある。この場合、モジュールの構成要素のことを「下位モジュール」あるいは「サブモジュール」と呼ぶ。

【0032】

- モジュールのエントリポイント

あるモジュールになんらかの処理を行わせるためのインタフェースのことである。(実装上はグローバル関数やグローバル手続きなどに相当する。)エントリポイントに適切なデータを渡して呼び出すと、モジュールは適切な処理を実行する

。呼び出されたモジュール側の処理が終了すると、処理結果がエントリポイントを呼び出した側(通常、別のモジュールである。)に渡される。

【 0 0 3 3 】

上記の一連の処理を「モジュールAのエントリポイントaを呼び出す。」と表現する。モジュールAが公開しているエントリポイントが1つしか場合は「モジュールAのエントリポイントを呼び出す。」あるいは「モジュールAを呼び出す。」と表現することができる。

【 0 0 3 4 】

- ゲートウェイ/ゲートウェイホスト

2つのコンピュータネットワークをつなげる場合、ゲートウェイホストを介してつなげることがある。ゲートウェイホストはコンピュータネットワーク間のデータのやり取りを制限するホストである。たとえば、2つのコンピュータネットワークAとBが接続されていると考える。ネットワークAに接続されているコンピュータaと、ネットワークBに接続されているコンピュータbで通信をおこなうとする。AとBの間にゲートウェイホストが設置されていない場合は、aとbは直接通信をおこなうことができる。しかし、AとBの間にゲートウェイホストGが設置されている場合は、

- 1、aはGと通信をおこなう。
- 2、aはGに対してbに通信内容を転送するように依頼する。
- 3、Gは通信内容を調べ、問題がなければbに転送する。という手順で通信をおこなわなければならない。

【 0 0 3 5 】

さらに、aから送られてきた内容に対して返答(リプライと呼ぶ)を行う場合は、

- 4、bはaにリプライを送るべく、Gとの通信を開始する。
- 5、aはGに対してbに通信内容を転送するように依頼する。
- 6、Gは通信内容を調べ、問題がなければaに転送する。という処理をおこなう必要がある。

【 0 0 3 6 】

ゲートウェイは通常のコンピュータをゲートウェイホストとして運用するためのソフトウェアである。ゲートウェイサーバとよぶこともある。

【0037】

- プロセス間通信/メッセージ/リプライ/ホスト名/ポート番号/ソケット

2つのプログラムが互いにデータをやりとりする時にはプロセス間通信機構が利用される。一般に、プロセス間通信機構はオペレーティングシステムに組み込まれている。特にソケットインタフェースを採用したプロセス間通信機構が広く利用されている。

【0038】

ソケットインタフェースを用いたプロセス間通信を行うためには、まず以下のデータを用意する。

1) 通信相手のソフトウェアが稼働しているコンピュータの名前(ホスト名)2) 通信相手のソフトウェアがプロセス間通信のために用意している「データの受け取り口」のID。

【0039】

この「データの受け取り口」のことを「ポート」と呼ぶ。また、「ポート」のIDのことを「ポート番号」と呼ぶ。

【0040】

ホスト名とポート番号を用意すれば、以下の手順でプロセス間通信を行うことができる。

1、ソケットを作成する。ソケットは、通信相手との間に成立した「通信路」に相当するものである。ただし、この時点では通信路は形だけのものである。そのまま通信に使えるわけではない。

2、ホスト名とポート番号を使ってソケットを初期化する。この処理によって、実際に通信路として機能する様になる。

【0041】

1と2の処理をあわせて「接続処理」と呼ぶ。接続処理に成功すると、ソケットを使ってデータを送信することが可能である。このときソケットをつかってやり取りされるデータのことを「メッセージ」と呼ぶ。メッセージの内容をあらわ

す言葉を添えて「要求メッセージ」、「返信メッセージ」と呼ぶこともある。また、「返信メッセージ」のことを「リプライメッセージ」あるいは「リプライ」と呼ぶこともある。

【0 0 4 2】

－ 通信プロトコル

プロセス間通信を行う時に、通信相手のプログラムは特定のパターンのメッセージしか受けつけないことがある。この、メッセージのパターンを規定したものが「通信プロトコル」である。単に「プロトコル」とも呼ぶ。標準的につかわれているものを以下に挙げる。実施例1でも以下のプロトコルを利用している。

【0 0 4 3】

－ ftpプロトコル

ftpはFile Transfer Protocolの略である。プログラム間でファイルをやり取りするためのプロトコルである。ftpプロトコルを処理する専用のサーバのことを「ftpサーバ」と呼ぶ。(ftpプロトコルの詳細はRFC765を参照のこと。)

【0 0 4 4】

－ httpプロトコル

WWW(World Wide Web)サービスのためのプロトコルである。主に、Webブラウザがサーバからデータを取得するために利用される。詳細はRFC1945およびRFC2068を参照されたい。

【0 0 4 5】

－ WWW(World Wide Web)サービス

WWWサーバないしWebサーバと呼ばれるサーバによって提供されるサービスである。利用者はInternet Explore(TM)などのWebブラウザを使って情報をWWWサーバから得ることができる。WWWサーバとWebブラウザはhttpプロトコルを利用してメッセージのやり取りを行う。このため、WWWサーバは「httpサーバ」と呼ばれることもある。

【0 0 4 6】

図6は第1の実施例の概要をあらわす図である。

【0 0 4 7】

カメラサーバ・ホスト(602)は、映像データをコンピュータネットワーク(610)を介して利用者に提供するサービスをおこなっている。

【0048】

カメラサーバ・ホスト(602)にはビデオカメラ(601)が接続されている。カメラサーバ・ホスト(602)はビデオカメラ(601)から画像を取得(キャプチャ)する。1フレームあたり10キロバイト程度の画像データを1秒間に1~10枚キャプチャすることが可能である。映像アップロード・ホスト(611)は、カメラサーバ・ホスト(602)のクライアントの1つである。10分ごとに1フレームなど定期的に、映像データをカメラサーバ・ホストから取得する。(一定時間分の動画を取得し、これを1かたまりのデータとして扱うこともできる。)

【0049】

映像データは、一般的なWWWホスト(606)に送信される。WWWホスト(606)は送信されてきた映像データをハードディスク等に蓄積し、利用者がWebブラウザをつかって参照できるようにする。図6では3人のユーザが他のホスト(607~609)でWebブラウザを利用している。

【0050】

映像アップロード・ホスト(611)は、映像データのアップロード処理と同時に、映像データに関する情報(映像情報データ)を作成し、アップロードすることができる。映像データを表示するためのHTMLファイルも、映像情報データの1つとみなせる。映像データごとに内容の違うHTMLファイルを作成することも可能である。

【0051】

映像アップロード・ホスト(611)は、WWWホスト(616)の様にモデム(613、614)を介した先のネットワーク(628)に接続されたWWWホストへも映像データをアップロードすることができる。また、映像アップロード・ホスト(611)は、WWWホスト(622)の様にゲートウェイ・ホスト(612)を介した先のネットワーク(627)に接続されたWWWホストへも映像データをアップロードすることができる。

【0052】

各WWWホスト(606、617、622)へのアップロード処理を同時に行うことも可能である。同一画像を3つのWWWホストにアップロードしたり、それぞれ異なったカメ



ラサーバホストから映像を取得することが可能である。

【 0 0 5 3 】

通常、アップロード・ホスト(611)は、定期的に映像アップロード処理を実行する様に設定される。ターミナル・クライアントを用いることで、設定とは無関係に即座に映像アップロードを実行させることができる。この機能は、定期的なアップロード処理の設定が、問題なく動作するか否かをテストするために使用される。図6では1人のユーザが他のホスト(612)上でターミナル・クライアントを起動している。

【 0 0 5 4 】

図7は実施例1が提供するユーザインタフェースの例である。701から708までは映像データの表示例である。図5と同様に、ウィンドウ(701)がオープンされている。ウィンドウ(701)は3つの表示領域(702、704、706)に分割されている。

【 0 0 5 5 】

表示領域(702)は、ページのタイトル(703)などあらかじめ用意できる情報を表示する。表示領域(704)は、閲覧可能な映像ファイルのリストが表示されている。各項目の文字列は、その映像がビデオカメラから取得された時刻をあらわしている。図7においては、午前5時(705)に取得された映像データが選択されている。

【 0 0 5 6 】

表示領域(706)には映像データ(707)と映像情報データ(708)が表示されている。映像情報データ(708)の内容は以下の通りである。

- 映像データ取得時の日付
- 映像データ取得時の時刻
- 映像データのデータサイズ
- ビデオカメラに関する情報(図ではカメラの向きをあらわすパン、チルト、ズームの値が表示されている。)

【 0 0 5 7 】

本実施例では、このように映像データによって変化する情報を利用者に提供することができる。

【 0 0 5 8 】

710以降はターミナルクライアントの表示例である。ウィンドウ(710)はtelnetなど一般的なターミナルプログラムの表示するウィンドウである。利用者は、ウィンドウ(710)に表示されているプロンプト(712)の後ろにコマンドをタイプする。コマンドは映像アップロードホストに送信されて処理される。処理結果はターミナルプログラムに返送され、ウィンドウ(710)に表示される。

【 0 0 5 9 】

図7では2つのコマンド

LIST (713) INFO 001 (715)

が実行され、各コマンドの結果(714、716)が表示されている。

【 0 0 6 0 】

LISTコマンドは、映像アップロードホストに設定されているアップロード設定がいくつされているかを問い合わせるためのコマンドである。結果(714)は4つのアップロード設定があり、それぞれ001、002、004、007というIDがつけられていることを示している。

【 0 0 6 1 】

INFO 001コマンドはアップロード設定001の内容を表示するよう要求するコマンドである。結果(716)は、カメラサーバ(IMAGE\_SERVER)が、「svcserv1.cano.co.jpというホストで、65310というポート番号でサービスをおこなっており、映像取得にはwvtcpプロトコルを用いる」ことをあらわしている。(wvtcpは、canoのWebView/Livescope(TM)カメラサーバから映像を取得するための通信プロトコルである。)

【 0 0 6 2 】

その他、以下のコマンドが使用可能である。

【 0 0 6 3 】

- ACTN ACTNは、以下の書式のコマンドである。

【 0 0 6 4 】

ACTN <タイマ設定オブジェクトID> このコマンドを映像アップロードサーバに送信すると、あらかじめ指定された時点に行われるべきアップロード処理を、指定時点を待たずに直ちに実行する。タイマ設定オブジェクトIDはこの指定時点に関する情報につけられたIDである。(詳細は後述する。)

【 0 0 6 5 】

このコマンドを実行すると、利用者にはアップロード処理の経過を示すメッセージが提示される。処理に成功したときは文字列"OK"を表示する。処理に失敗したときは文字列"ERROR"と失敗の理由を表示する。

【0 0 6 6】

以下にACTNを使用した例をしめす。

> LIST ITEM-COUNT=4

001、002、004、007 OK

【0 0 6 7】 > INFO 001

IMAGE-SERVER: wvtcp://svcserv1.xpc.cano.co.jp:65310

IMAGE-DATA: ftp://svcserv3.xpc.cano.co.jp:21/Upload/images/image\_001.jpg

INFO-DATA: ftp://svcserv3.xpc.cano.co.jp:21/Upload/images/info\_of\_image\_001.txt

OK

【0 0 6 8】 > ACTN 001

Capturing Image from: wvtcp://svcserv1.xpc.cano.co.jp:65310 .... done.

Uploading an Image File: ftp://svcserv3.xpc.cano.co.jp:21/Upload/images/image\_001.jpg ... done.

Uploading an Info File: ftp://svcserv3.xpc.cano.co.jp:21/Upload/images/info\_of\_image\_001.txt ...done.

OK

【0 0 6 9】 > ACTN 004

Capturing Image from: wvtcp-vc://svcserv1.xpc.cano.co.jp:65310 .... done.

Uploading an Image File: ftp://svcserv4.xpc.cano.co.jp:21/Upload/images/image\_001.jpg .....

ERROR

Failed to upload an image file: ftp://svcserv4.xpc.cano.co.jp:21/Upload/images/image\_00.jpg

Failed to connect to the server: ftp://svcserv4.xpc.cano.co.jp:21

【0070】

上記の例では、LISTコマンドで利用可能なタイマ設定のリストを表示させ、表示されたタイマ設定のうちの2つをACTNコマンドで実行させている。

【0071】

LISTコマンドのリプライはまず先頭に表示されるタイマ番号の個数が以下の書式で表示される。

【0072】

ITEM-COUNT=<タイマ番号の個数> 上記の例では、タイマ番号の個数は'4'である。

【0073】

ITEM-COUNTの次に、使用できるタイマ番号が1つずつ表示される。上記の例では 001、 002、 004、 007が表示されている。最後に、LISTコマンドの実行に成功したことをあらわすために「OK」が表示される。

INFOコマンドではタイマ番号'001'の設定内容が表示される。

【0074】

IMAGE-SERVER: 「映像取得設定」

IMAGE-FILE: 「画像データの送信先」

INFO-FILE: 「画像情報データの送信先」

「映像取得設定」は以下の書式の文字列である。

【0075】

<プロトコル名>://<ホスト名>:<ポート番号> "プロトコル名"は映像アップロードサーバがカメラから映像データを取得するために使用されるプロトコルの名前である。たとえば、カメラサーバがhttpプロトコルによる映像取得インタフェースを提供しているならば、プロトコル名に"http"を指定することができる。"ホスト名"はカメラサーバ・ホストのホスト名である。"ポート番号"はカメラサーバにメッセージを送信するために利用することのできるTCPポート番号である。WebView/Livescopeカメラサーバを利用した場合は'65310'がデフォルトのポート番号となる。

【0076】

「画像データの送信先」および「画像情報データの送信先」は以下の文字列である。

【0077】

〈プロトコル名〉: // 〈ホスト名〉: 〈ポート番号〉/ 〈ファイルのパス〉 “プロトコル名”はアップロードサーバがファイルをアップロードするために使用するプロトコルの名前である。通常はFTPサーバと通信することでファイルをアップロードするので“ftp”が使用される事が多い。

【0078】

“ホスト名”はWWWホストのホスト名。“ポート番号”はWWWホスト上で送信先となるサーバのポート番号。通常はFTPサーバのポート番号を使用する。“ファイルのパス”は、送信データの置き場所(ファイル名)である。

【0079】

これらの書式はURL(Universal Resource Locator)表記に準じている。(URLの詳細についてはRFC1738を参照のこと。) 【0080】 最初のACTNコマンドではタイマ番号'001'が実行される。映像取得から画像ファイル、画像情報ファイルのアップロードおよび画像情報リストの更新がすべて正常終了している。

【0081】

2つめのACTNコマンドではタイマ番号'004'が指定されている。この場合は、映像取得には成功したが、画像ファイルのアップロードに失敗している。このためERRORが表示される。理由は“Failed to connect to the server: ftp://svcserv4.xpc.cano.co.jp:21”と表示されているとおり、アップロード先コンピュータのFTPサーバとの接続処理に失敗したためである。

【0082】

図8はテンプレートデータとテンプレートデータから作成された映像情報データの例である。テンプレートデータから映像情報データをつくる際には、まずテンプレートデータからキーワードを探し出し、キーワードを実際の映像情報で置き換えることで映像情報データを作成する。

【0083】

本実施例では始めと終わりを” @@”で囲まれた文字列をキーワードとして処理する。図8においては802、803、804、805がキーワードである。801はキーワードではない。このため、806に示されるように変更されることなく映像情報データに組み込まれる。802と805のキーワードはそれぞれ「IMAGE\_CAPTION」、「IMAGE\_FILE」

であり、そのまま

映像を説明するための文字列「富士山の映像」(807) 映像データのWWWホスト上でのファイル名「mt\_Fuji.jpg」(810)で置換されている。

【 0 0 8 4 】

803、804には同一のキーワード DATE\_STAMP が指定されている。DATE\_STAMPは映像が取得された日時をあらわすキーワードである。しかし、キーワードの直後に’[と]’で囲まれた文字列がおかれており、803と804ではその内容が異なっている。[]で囲まれた文字列は、キーワードを実際の値で置き換えるときの書式を指定したものである。

【 0 0 8 5 】

803の書式指定は「%Y年 %m月 %d日 (%a)」となっている。%Yは年、%mは月、%dは日、%aは曜日を表示させるための指定である。その結果は808に示すように「1999年 1月 21日 (火)」という文字列に置き換えられている。

【 0 0 8 6 】

また、804の書式指定は「%H:%M」と指定されている。%Hは時間、%Mは分を表示させるための指定である。その結果809に示すように「午前 10:00」と表示される。

【 0 0 8 7 】

以下に指定可能なキーワードとその意味を列挙する。これらは、映像取得スケジュールで設定されている、日 時や、カメラサーバの所在地、装置の設定、カメラの制御情報から得ることができる。

【 0 0 8 8 】

@@IMAGE_FILE@@	// 映像データにつけられた名前
@@CHAR_SET@@	// 作成された映像情報データの文字コード

```

// (ASCII、 JIS、 SJIS、 EUC、
// UNICODE、 UTF-8のいずれか)      @@TIMER_CA
PTION@@ // 時点指定説明用の文章
@@DATE_STAMP@@ // 映像取得の日付&時刻
@@LANGUAGE @@ //作成された映像情報データの言語
// (English、 Japanese、 etc...)      @@TIME_
STAMP@@ // 映像取得のタイムスタンプ(1970年からの秒数)      @@IMAGE_
CAPTION@@ // 映像データ説明用の文章
@@PAN@@ // カメラの左右の向き
@@TILT@@ // カメラの上下の向き
@@ZOOM@@ // カメラのズーム倍率
@@IMAGE_WIDTH@@ // 映像データの横サイズ
@@IMAGE_HEIGHT@@ // 映像データの縦サイズ
@@TARGET_HOST@@ // 送信先のホスト名
@@HTTP_HOST_AND_PORT@@ // httpサーバのホスト名とポート番号
@@CAMERA_SERVER_HOST@@ // カメラサーバのホスト名
@@SEQ_NUMBER@@ // 映像取得処理の実行回数
@@INFO_FILE@@ // 映像情報データにつけられた名前

```

【 0 0 8 9 】

以上のような書式をテンプレートに設定しておくことによって、たとえば、WWWサーバが、ネットワーク上の閲覧者から、映像データを送信するよう要求を受けたら、WWWサーバは、上記のようなテンプレートに、映像取得時の情報等を自動的に埋め込んで、HTMLやXMLといった、表示用映像制御データを送信することが出来る。

【 0 0 9 0 】

つまり、HTML文書等の表示用映像制御データを、いちいち手動で更新すること無く、又、映像に写し込むことなく、ネットワーク上のユーザーに、映像データ取得時の取得日時や映像データのサイズ、タイトル、パン・チルト・ズーム等のカメラ制御情報を付加した映像データを提供することが出来る。

【 0 0 9 1 】

・ 構成図の説明

図2は実施例のハードウェア構成をあらわす図である。図においてコンピュータ(201)は映像アップロードホストである。映像アップロードホスト(201)はCPU(202)とメイン・メモリ(203)から構成されている。CPU(202)とメイン・メモリ(203)間はバス(204)で接続されている。

【 0 0 9 2 】

メインメモリ(203)内には、アップロード処理を実際に実行する映像アップロードサーバ(205)がロードされている。映像アップロードホスト(201)には、RS232C等の外部インタフェース(209)を介してディスプレイ(206)、キーボード(207)、マウス(208)、モデム(221)、が設置されている。(ただし本実施例においては206、207、208は必須ではない。その場合、コンピュータネットワーク(212)を介して別のコンピュータから操作をおこなう必要がある。

【 0 0 9 3 】

映像アップロードホスト(201)には2次記憶装置(ハードディスク)(210)が設置されている。ハードディスク(210)上には映像アップロードサーバ(205)の初期化処理のための各種情報を含んだ各種の設定ファイル群(211)が蓄積されている。

【 0 0 9 4 】

コンピュータネットワーク(212)によって映像アップロードホスト(201)とカメラサーバ・ホスト(213)、WWWホスト(215)、その他のホスト(218、219、220)は接続されている。

【 0 0 9 5 】

カメラサーバホスト(213)上ではカメラサーバ(214)が稼働している。カメラサーバ(214)はビデオカメラから映像データを取得して映像アップロードサーバ(205)に送信するサーバ・プログラムである。カメラサーバホスト(213)のハードウェア構成は、映像アップロードサーバ(201)と同じである。このため、カメラサーバ(214)を映像アップロードホスト(201)上で稼働させるような運用形態も可能である。

【 0 0 9 6 】



WWWホスト(215)ではhttpサーバ(216)とftpサーバ(217)が稼働している。httpサーバ(216)はWWWホストのメモリやハードディスク等の記憶装置に蓄積されているデータをWebブラウザで参照することができるようにする機能を持つサーバ・プログラムである。図2では、クライアントホスト(218、219、220)上でWebブラウザが利用されているものとする。httpサーバとWebブラウザはhttpプロトコルを用いて通信をおこなう。

## 【0097】

ftpサーバ(217)は映像アップロードサーバ(205)から送信されてくる映像データを、WWWホスト(215)の記憶装置に蓄積する機能を提供するサーバ・プログラムである。映像アップロードサーバ(205)とftpサーバ(217)はftpプロトコルを用いて通信を行う。(httpサーバによっては、ftpサーバと同じ機能を持つものも存在する。このようなhttpサーバを利用する場合はftpサーバは不要である。) WWWホスト(215)のハードウェア構成は、映像アップロードサーバ(201)と同じである。このため、httpサーバ(216)とftpサーバ(217)を映像アップロードホスト(201)上やカメラサーバホスト(213)上で稼働させるような運用形態も可能である。

## 【0098】

その他のホスト(218、219、220)はWWWブラウザを動作させるためのコンピュータである。構成上は映像アップロードホスト(201)と同じであるが、上記のサーバ(205、214、216、217)を動作させる能力は必要ない。したがって、比較的低速度のCPUやバス、小容量のメインメモリ、マウス、キーボード、2次記憶装置を持たないような構成のコンピュータでも構わない。また、図2では3つのその他のホストがかかっているが、実際にはその数はいくつでも構わない。

## 【0099】

なお、インターネットを介して映像データを送受信する場合、映像データファイルは、HTML形式でやり取りするのが、現在のところもっとも効果が有るが、多様なアプリケーションでの互換性が高いXML形式で記述するようにすれば、よりデータの互換性が高まって便利である。

## 【0100】

図1は映像アップロードサーバ(101)の内部構成をあらわす図である。

【 0 1 0 1 】

映像アップロードサーバ(101)は、以下のソフトウェアモジュールから構成されている。

- OSインタフェース・モジュール(102)- タイマ処理モジュール(103)- アップロード処理モジュール(107)- サーバ通信管理モジュール(117)- ターミナル処理モジュール(125)

【 0 1 0 2 】

●OSインタフェースモジュール(102) OS(オペレーティングシステム)の機能を利用するための、インタフェースを提供するモジュールである。タイマ処理モジュール(103)など他のモジュールは、OSインタフェースモジュール(102)を介してOSの機能を利用することができる。本実施例においては、ファイル操作やプロセス間通信などがOSインタフェースモジュール(102)を利用して処理される。

【 0 1 0 3 】

●タイマ処理モジュール(103) 適当な時点を設定しておく、その時点になったときに処理を開始するモジュールである。映像アップロードサーバにおいては、以下の3つの処理がタイマ処理から開始される。

- アップロード処理
- データの有効期限が切れた時の、データの削除処理
- 映像および映像情報データ送信処理

【 0 1 0 4 】

上記の処理、以下のサブモジュールによって管理される。

- アップロードタイマ管理サブモジュール(104)- データ有効期限タイマ管理サブモジュール(106)- データ送信タイマ管理サブモジュール(105)

【 0 1 0 5 】

その対応は以下の通りである。

- アップロードタイマ管理サブモジュールの呼び出し先はアップロード処理モジュール(107)のアップロード処理(110～114に相当する)- データ送信タイマ管理モジュール(105)の呼び出し先はデータ送信サブモジュール(112)のデータ送信処理
- データ有効期限タイマ管理サブモジュール(106)の呼び出し先はデータ削除処理サブモジュール(115)のデータ削除処理

【0 1 0 6】

●アップロード処理モジュール(107) 映像データおよび映像情報データのアップロード処理を実施するモジュールである。本実施例におけるアップロード処理は以下のステップから成り立っている。

- 映像データ取得処理 -- カメラサーバから映像データを取得する。
- 映像データ加工処理 -- 取得した映像データのサイズを変更する/画像データを合成するなど、ユーザーの指示やあらかじめの設定に基づいて、映像データの加工を行う。

【0 1 0 7】

加工処理の例としては以下のものがある。

- 映像データに日付やメッセージを埋め込む
- 複数の映像データをつなぎあわせてパノラマ画像を作成する。
- 映像サイズを縮小してサムネイル用データを作成する。

【0 1 0 8】

- 映像データ送信処理 -- 加工した映像データをWWWホストに送信する。
- 映像情報データ作成処理 -- 映像データに関する情報を含んだ映像情報データを作成する。

【0 1 0 9】

なお、映像情報には、映像データのサイズ/取得の日時と時刻/取得時のカメラの向きなどの情報が含まれる。本実施例では、映像情報データはテキストデータとして作成される。(ただし、他のタイプのデータとして作成することも可能である。以降の説明はテキストデータ以外のデータ作成にもそのまま当てはまる。)

【0 1 1 0】

映像映像情報データ作成処理は以下の2つのステップからなりたっている。

【0 1 1 1】

- テンプレート・データ取得処理

映像情報データのひな形となるテンプレートデータを取得する。通常、テンプレートデータは、映像アップロードホストの2次記憶装置内にファイルとして保

存されている。(テンプレートデータをカメラサーバや WWWホスト側で保持するような構成も可能である。) 【0112】 - キーワード置き換え処理(狭義の映像映像情報データ作成処理) ここでは以下の処理によって映像情報データを作成する。1)まず、あらかじめ決められたキーワードがテンプレートデータ内に埋め込まれているか調べる。2)キーワードが見つかったら、その部分を映像サイズ等の値でおきかえる。3)この処理をテンプレートデータの最後までおこなう。この処理の詳細については図19以降の解説の時に改めて説明する。

【0113】

- 映像情報データ送信処理

作成した映像情報データをWWWホストに送信する。

【0114】

- データ削除処理

WWWホストに送信したデータを削除する要求を、WWWホストに送信する。

【0115】

なお、以下のサブモジュールはそれぞれセミコロン以下の処理を実施する。

- 映像データ取得サブモジュール(110) ; 映像データ取得処理
- 映像データ加工サブモジュール(111) ; 映像データ加工処理
- データ送信サブモジュール(112) ; 映像データ送信処理
- テンプレート・データ取得サブモジュール(113) ; テンプレート・データ取得処理
- 映像情報データ作成サブモジュール(114) ; 映像情報データ作成処理
- データ削除サブモジュール(115) ; データ削除処理
- データ送信サブモジュール(112) ; 映像データ送信処理、映像情報データ送信処理

【0116】

1つのアップロード処理をどのように実施するかは、上記のサブモジュールをどのように組み合わせて呼び出すかによって決まる。この組み合わせを記述したものを「アップロード設定」とよぶ。(「アップロード設定」は各種設定情報群(129)の一部である。各種設定情報群(129)については後述する。) 【011

7]

アップロードタイマ管理サブモジュール(104)から処理開始を通知されると、アップロード処理サブモジュール(107)はアップロード設定の内容にしたがい、各サブモジュール(110～114)の処理を行わせる。

【0 1 1 8】

映像情報データベース(116)は、映像データおよび映像データに関する情報を一時的に保持するためのデータベースである。これらの情報は映像データ取得サブモジュール(110)や映像データ加工サブモジュール(111)によって設定される。設定された情報は映像情報データ作成サブモジュール(114)が映像情報データを作成するときなどに使用される。

【0 1 1 9】

●サーバ通信処理モジュール(117) 映像データ取得サブモジュール(110)が映像データを取得し、データ送信サブモジュール(112)がデータを送信するためには、それぞれカメラサーバやWWWホストとネットワークを介してプロセス間通信をおこなわなければならない。サーバ通信処理モジュール(117)は、カメラサーバやWWWホスト(すなわちhttpサーバやftpサーバ)との通信処理を実施するモジュールである。このモジュールも複数のサブモジュールから構成されている。

【0 1 2 0】

- サーバ接続処理サブモジュール(118) カメラサーバ、httpサーバ、ftpサーバなどのサーバと通信するために必要な情報(サーバ情報)を管理し、サーバとの接続処理を制御するモジュールである。 【0 1 2 1】

- 経路接続処理サブモジュール(119) サーバ情報で指定されたサーバと実際に通信するには、そのサーバとどの経路で通信が可能であるかに関する情報が必要である。この情報を「経路情報」と呼ぶ。経路情報は以下の項目からなりたっている。

- モデム関連情報(利用可能なモデムポート、相手先の電話番号など)- ゲートウェイ関連情報(ゲートウェイ・ホストの名前、受けつけるプロトコルの種類など)

【0 1 2 2】

経路接続処理サブモジュール(119)は、経路情報を管理し、指定された経路に

よる接続処理を制御するモジュールである。

【0 1 2 3】

サーバとの接続処理が成功したならば、通信相手のサーバが受けつけることのできる通信プロトコルを用いて情報のやり取りをおこなうことができる。

【0 1 2 4】

- ftpプロトコル処理サブモジュール(120) ftpプロトコル処理サブモジュール(120)はftpプロトコルを使ったデータ通信を処理するモジュールである。主に、映像データや映像情報データのWWWホストへの送信のために利用される。

【0 1 2 5】

- httpプロトコル処理サブモジュール(121) httpプロトコルを使ったデータ通信を処理するモジュールである。主に、テンプレートデータの取得のために使用される。また、映像データや映像情報データのWWWホストの送信にも使用されることがある。

【0 1 2 6】

- wvtcpプロトコル処理サブモジュール(123) wvtcpプロトコルは、CANO WebView/Livescopeカメラサーバ(TM)から映像を取得するためのプロトコルである。wvtcpプロトコル処理サブモジュールは、wvtcpによる映像データ取得のために使用されるモジュールである。

【0 1 2 7】

カメラサーバとしてWebView/Livescope以外のカメラサーバを利用したり、Gopherプロトコルを用いて映像データの送信を行う場合は、専用のプロトコル処理モジュールを追加することになる。その場合でも以下の実施例の実現方式がそのまま適用可能である。

【0 1 2 8】

●ターミナル処理モジュール(125) ターミナル処理モジュール(125)は、図7で説明したようなターミナル・クライアントによるユーザ操作を処理するためのモジュールである。

【0 1 2 9】

ユーザがターミナル・クライアントで入力したコマンドは、まずコマンド受信

サブモジュール(126)で受信される。ターミナル・クライアントとコマンド受信サブモジュール間の通信もOSインタフェース・モジュール(102)が提供するプロセス間通信の機構を用いて実現されている。コマンド受信サブモジュール(126)で受信されたコマンドは、コマンド実行サブモジュール(127)で解釈され実行される。(この「実行」は、アップロード処理モジュール(107)およびそのサブモジュール(110~114)に適切なパラメータを渡して呼び出すだけである。タイマ処理モジュール等による呼び出しとまったく同じやり方で処理される。詳細はタイマ処理モジュールによる呼び出し処理を参照されたい。) 実行結果は、リプライ送信サブモジュール(128)によってターミナル・クライアントに送り返される。ターミナル・クライアントは図7で示したように、結果を表示する。

#### 【 0 1 3 0 】

●各種設定情報群(129) 各種設定情報群(129)は、タイマ設定/アップロード設定/サーバ情報/経路情報などの集合である。各種設定情報群の内容については図3で説明する。

#### 【 0 1 3 1 】

図3は各種設定情報群(図 1 における 1 29)の詳細を説明するための図であり、特に、アップロード処理を行う場合を示している。

#### 【 0 1 3 2 】

この各種設定情報群(301)は映像アップロードサーバ(101)の一部として各モジュールによって利用される。これらの情報は映像アップロードサーバ(101)の起動時にハードディスク(210)内の設定ファイル群(211)内の情報で初期化される。以下ではファイルに書かれている状態の設定情報のことを「設定ファイル」と呼ぶ。映像アップロードサーバ(101)内に読みこまれた状態の設定情報のことを「設定オブジェクト」と呼ぶ。

#### 【 0 1 3 3 】

各種設定情報群(301)は以下の設定オブジェクトから構成されている。

- タイマ設定オブジェクト(302)- アップロード設定オブジェクト(303)- 映像データ取得設定オブジェクト(304)- 映像情報データ設定オブジェクト(305) - データ送信設定オブジェクト(306)- サーバ設定オブジェクト(307)- 経路設定オブ

ジェクト(308) 【0 1 3 4】

なお、図中の矢印は、各オブジェクトに矢印先のオブジェクトに関するID情報を含んでいることを示す。

【0 1 3 5】

●タイマ設定オブジェクト(302) タイマ設定オブジェクト(302)はタイマ処理モジュール(103)に使用される設定オブジェクトである。タイマ処理モジュール(103)側の処理により、タイマ設定オブジェクトには以下の3種類のタイプが用意されている。

アップロードタイマ設定タイプ

データ送信タイマ設定タイプ

データ有効期限タイマ設定タイプ

【0 1 3 6】

それぞれ、アップロードタイマ管理サブモジュール(104)、データ送信時刻設定サブモジュール(105)、データ有効期限タイマ管理サブモジュール(106)の時点設定に使用されるものである。

【0 1 3 7】

タイマ設定オブジェクトを構成する属性として、以下のものを格納している。

【0 1 3 8】

- タイプ属性

アップロードタイマ設定タイプ、データ送信タイマ設定タイプ、データ有効期限タイマ設定ファイルの3種類のタイプのうち、どのタイプのタイマ設定であるかを指定するための設定項目である。

【0 1 3 9】

upload /copy/fileの3つの値のうちの1つを指定することができる。uploadはアップロードタイマ設定タイプであることを示す。copyはデータ送信タイマ設定タイプであることを示す。fileはデータ有効期限タイマ設定ファイルであることを示す。

【0 1 4 0】

- 時点設定属性



アップロード処理などを起動すべき時点を指定する。「日付設定」、「曜日設定」、「時刻設定」の3つの項目から成り立っている。日付設定は指定したい時点の日付の部分を指定することができる。指定方法は以下の3種類である。

【0 1 4 1】

方法1) 単日付指定

ある日付を指定する。

書式 -- <西暦年>:<月>:<日付>例。 1999:01:20 この例は1999年の1月20日を指定したものである。

この指定方式が、日付指定の基本となる。

【0 1 4 2】

方法2) 期間日付指定

ある日付から別の日付の間の期間を指定する。

書式 -- <開始日の単日付指定>-<終了日の単日付指定>例。 1999:01:01-1999:12:31この例は、1999年の1月1日から12月31日までの期間を指定したものである。

【0 1 4 3】

方法3) 間隔付き日付指定

期間日付指定は毎日が指定日付となるが、間隔付き日付指定では

1日おきや2日おきに処理を行わせることができる。書式 -- <期間日付指定>/<月間隔>:<日間隔>例1、 1999:01:01-1999:12:31/00:00:02例2、 1999:01:01-1999:12:31/00:01:00例3、 2000:01:01-2999:12:31/01:00:00例1は「1999年1月1日から12月31までの2日おき」を指定したものである。

例2は「1999年1月1日から12月31までの1ヶ月おき」を指定したものである。

例3は「2000年1月1日から2999年12月31までの1年おき」を指定したものである。

【0 1 4 4】

なお、内部的には、すべての日付指定は「間隔付き日付指定」として処理される。すなわち「期間日付指定」は日間隔がゼロである間隔付き日付指定としてあつかうことができる。また、「単日付指定」は開始日と終了日が等しい間隔付き日付指定として処理することが可能である。

【0 1 4 5】

時刻指定も同様にして「単時刻指定」、「期間時刻指定」、「間隔付き日付指定」が可能である。指定時の書式は日付指定と同様である。以下に例を示す。

例1、 13:00例2、 07:00-18:00例3、 07:00-18:00/00:30例4、 05:00-20:00/02:00例1は「午後 1 時」をあらわす。

例2は「午前 7 時から午後 6 時の毎分」をあらわす。

例3は「午前 7 時から午後 6 時までの 3 0 分ごと」をあらわす。

例4は「午前 5 時から午後 8 時までの 2 時間ごと」をあらわす。

【0 1 4 6】 曜日指定の場合は、指定したい曜日を列挙する。以下に例を示す。

【0 1 4 7】

月、水、金

上記の例は「月曜日、水曜日、金曜日」をあらわしている。

【0 1 4 8】

- 関連設定オブジェクトIDのリスト

タイマ処理モジュールで指定された時点になったときに、定められた処理(アップロード処理やデータ送信処理)を行うために、オブジェクト間でオブジェクトのIDの受け渡しを行って、関連するモジュールを呼び出す。呼び出すべき設定オブジェクトのオブジェクトIDを、前述のタイプ属性によって決めたリストが、関連設定オブジェクトIDのリストである。

【0 1 4 9】

例えば、タイマー設定オブジェクトの設定が、アップロードタイマ設定タイプなら、アップロード設定オブジェクト(303)を示すID、データ送信タイマ設定タイプならばデータ送信設定オブジェクト(306)を示すID、データ有効期限タイマ設定タイプならば有効期限付きデータ設定オブジェクト(309)を示すIDを保持しているリストである。

【0 1 5 0】

例えば、図3においては、タイマ設定オブジェクト(302)がアップロードタイマ設定タイプであるので、このリストに基づいて、アップロード設定オブジェクト(303)を呼び出すことになり、この関係が、302から303への矢印として表現され

ている。

#### 【0151】

本実施例では複数のアップロード処理が指定されていた場合、各アップロード処理を順番に1つずつ実行する。(各アップロード処理を同時に実行させることも可能である。) 【0152】

●アップロード設定オブジェクト(303) アップロード設定オブジェクトはアップロード処理の内容を指定するための設定オブジェクトである。どのように映像データを取得し、そのどのようにWWWホストに送信するかを指定する。これらの情報はそれぞれ映像データ取得設定オブジェクト(304)とデータ送信設定オブジェクト(306)内に保持されており、アップロード設定オブジェクト内(303)にはこれらの設定オブジェクトのオブジェクトIDを指定することになる。

#### 【0153】

ただし、1つのアップロード設定オブジェクト(303)が保持することができる映像データ取得設定オブジェクト(304)は1つだけである(図中の1:1はこれを示す)。これに対し、データ送信設定オブジェクト(306)は複数指定することができる(図中の1:nはこれを示す)。これは、1つの映像データを複数のWWWホストに送信できることを反映した構造である。

#### 【0154】

●映像取得設定オブジェクト(304) 映像データ取得のために必要な、以下のような属性値を保持するために使用される。

#### 【0155】

- 映像パラメータ(リモート映像パラメータ) カメラサーバのよって取得する映像データに対してパラメータを設定することができる。本実施例ではWebView/Livescopeカメラサーバを利用するので、映像データ取得時に映像データのサイズやビデオカメラの向きを指定することができる。これらの情報を映像パラメータと呼ぶ。(次に述べる「ローカル映像パラメータ」と区別するために、「リモート映像パラメータ」と呼ぶこともある。) 【0156】

映像パラメータ

カメラサーバが映像パラメータ指定の手段を提供していない場合でも、映像に

関する情報をあらかじめ映像アップロードサーバ(101)側で用意することが考えられる。例えばカメラの向きが変更できないカメラサーバならば、あらかじめカメラの向きに関する情報を映像アップロードサーバ(101)側に用意しておけばよい。また、映像データの内容説明用のテキストデータなど一般にカメラサーバでは提供していない情報も考えられる。これらの情報はローカル映像パラメータとしてアップロード設定ファイルで設定することができる。

## 【0157】

ーサーバ設定オブジェクトのオブジェクトID 映像データの取得先となるカメラサーバとの通信処理に使用されるサーバ設定オブジェクトのIDである。

## 【0158】

ーデータ送信設定オブジェクトのオブジェクトIDリスト

データ送信設定オブジェクトのIDを複数保持する。これらのオブジェクトはWWWホストに映像データを送信するときに参照される。

## 【0159】

●データ送信設定オブジェクト(306) 映像データや映像情報データを、WWWホストに送信するために使用される設定オブジェクトである。送信するデータが2つのタイプの別れることから、データ送信設定オブジェクトも2種類のタイプが用意されている。

## 【0160】

映像データ送信設定タイプ

映像情報データ送信設定タイプ

## 【0161】

タイプの区別はタイマ設定オブジェクト(302)と同様にタイプ属性で行う。映像データ送信設定タイプのオブジェクトは、データ送信だけでなく映像データ加工に関する情報を保持することを特徴としている。また、映像情報データ設定オブジェクト(305)のIDを複数保持することが可能である。このようにすることで、データの送信先であるWWWサーバに合わせて独自のデータの加工や映像情報データを生成させることが可能である。

## 【0162】

また、映像データ送信設定タイプでは「映像加工パラメータ」を属性値としてもつことができる。このパラメータは、カメラサーバから取得した映像データに対し、映像データ加工サブモジュール(111)で実行すべき処理を指定する。具体的には、映像加工のために呼び出すべき外部コマンド(プログラム)と、呼び出すときの引数の書式を指定すればよい。映像加工の例としては映像サイズの縮小や他の映像データとの合成が考えられる。

## 【0163】

それ以外では、2つのタイプの内容に違いはない。以下に共通な属性を列挙する。

## 【0164】

- サーバ設定オブジェクトのID データの送信先であるHTTPサーバやFTPサーバに関するサーバ設定オブジェクトのIDを保持する。

## 【0165】

- 送信パラメータ

データ送信時に付加される情報である。通常は、データの種別、データの置き場所となるファイルやディレクトリの名前を指定する。また、「送信したデータで古いデータを置き換える」、「送信データを古いデータの最後尾に追加する。」といった受信側でのデータ蓄積方式の指定もここで行う。

## 【0166】

- 映像情報データ設定オブジェクトのIDのリスト

映像情報データ設定オブジェクトのIDリストを保持する。ここで指定された映像情報データ設定オブジェクトは映像情報データを作成するために使用される。

## 【0167】

- 一括送信フラグ

データ送信設定オブジェクトはデータ送信タイマ設定タイプのタイマー設定オブジェクトから参照されることがある。これは一括送信方式を実現するためである。通常、データ送信処理はアップロード処理の最終段階として、データの加工や作成処理に続けて処理される。これに対し、一括送信方式では、データ送信処理をしばらく保留し、あらかじめ設定された時点になったときに実行する方式で

ある。一括送信フラグがONになっているときは、このデータ設定オブジェクトが一括送信処理の対象になっていることを意味する。

【0168】

●映像情報データ設定オブジェクト(305) 映像情報データを作成するために必要な情報を保持するための設定オブジェクトである。映像情報データのひな形となるテンプレートファイルの置き場所に関する情報や、生成すべきデータの種別を保持する。具体的には以下の通りである。

【0169】

- テンプレートファイル名

情報データのテンプレートデータを保持しているファイルのファイル名である。

【0170】

- データ作成パラメータ

情報データ作成時に使用されるパラメータである。本実施例では情報データをテキストデータとして作成する。データ作成パラメータとしてはテキストデータの言語およびコード系の指定が使用される。

【0171】

- 送信パラメータ

データ送信設定オブジェクト内の「送信パラメータ」属性と同じ内容である。映像情報データはこの内容に基づいてWWWホストに送信される。ただし、送信先はデータ送信設定オブジェクト(306)側に指定されているため、映像情報データ設定オブジェクトでは指定できない。

【0172】

●有効期限つきデータ設定オブジェクト(309) 映像アップロードサーバ(101)は、WWWホストへのデータ送信処理の実行の後、一定の期間が経過した時点でWWWホストへそのデータの削除を要求する処理をおこなうことができる。この機能は、記憶媒体の記憶容量の不足を防止するために、アップロードしたデータの有効期限を制限するために使用される。有効期限つきデータ設定オブジェクト(309)はこのデータ削除処理を行うために削除の対象となるデータに関する情報を保持

する。また、WWWホストにデータの削除処理を行わせるために、サーバ設定オブジェクトへの参照も一つだけ設定することができる。

【0173】

なお、有効期限まで映像データを蓄積しておくのではなく、古い映像ファイルを新しい映像ファイルに上書きする方法もあるが、この方法の欠点は、過去の映像データが上書きで失われてしまうことである。

【0174】

●サーバ設定オブジェクト(307) カメラサーバやWWWホストなど通信相手のサーバに関する情報を保持するための設定オブジェクトである。具体的には以下の属性値を保持している。

- ホスト名 — 相手のサーバが稼働しているホストの名前である。
  - ポート番号 — サーバがサービスのインタフェースとしているTCPポートの番号である。
  - プロトコル名 — サーバが受信することのできるプロトコル名である。
  - 接続パラメータ — サーバとの通信のために必要なパラメータである。
- 例えばftpサーバではサーバとの通信を行う時にユーザ名とパスワードを要求する。このため、ftpサーバが相手のときはユーザ名とパスワードを接続パラメータとして登録しておく必要がある。

【0175】

- 経路設定オブジェクトのオブジェクトIDのリスト

サーバとの接続に必要な経路情報を保持している経路設定オブジェクトのIDのリストである。ただし、これら経路設定オブジェクトを同時に使用することはできない。その内の1つだけが使用可能である。ある経路での通信にトラブルが発生した時に、別の経路を使って再度通信を試みるといった処理をおこなうことは可能である。

【0176】

●経路設定オブジェクト(308) カメラサーバやWWWホストと通信を行うために必要なモデムやゲートウェイに関する情報を保持するための設定オブジェクトである。これによってインターネットやLANでの利用に、対応することができる

内容は以下のとおりである。

－ モデムポート ― モデムと映像アップロードホストを接続するためのモデムポートの指定である。

－ 電話番号 ― 相手先モデムの電話番号

－ モデム接続パラメータ ― に必要なパラメータである。

通常はユーザ名とパスワードをモデム接続パラメータとして登録する。

－ ゲートウェイ・ホスト名 ― ゲートウェイとして使用されているホストの名前。

－ ゲートウェイ・ポート番号 ― ゲートウェイがサービスを行っているTCPポート番号。

－ プロトコル名 ― ゲートウェイが受理することのできるプロトコルの名前。

#### 【0 1 7 7】

##### フローチャートの説明

図9は設定ファイルから設定オブジェクトを作成する工程をフローチャートにあらわしたものである。図3をみればわかるように、設定オブジェクトはタイマー設定情報から順に参照関係で結ばれている。ちょうどタイマー設定情報をつまみあげれば、他の設定オブジェクトもズルズルとついてくるような構成になっている。したがって、例えば設定オブジェクトの初期化処理もタイマー設定情報の初期化から開始し、順にその参照先の設定オブジェクトを初期化してゆくことによって実施される。

#### 【0 1 7 8】

S9a01からS9a05まではタイマー設定オブジェクトの作成・初期化処理をあらわしている。まず、タイマ設定ファイルの内容をメモリ上に読み出す(S9a01)。つぎに、タイマ設定オブジェクトのオブジェクト領域をメインメモリ上に確保する(S9a02)。この時点でのタイマ設定オブジェクトには実際のタイマ設定情報は保持されていない。メモリ上に確保された単なるデータの保持領域にすぎない。そこで、タイマ設定ファイルから読み出した各項目の値をタイマ設定オブジェクトにセットする(S9a03)。具体的には、タイプ属性/日付指定属性/曜日指定属性/時刻指定属性の値を取り出して、タイマ設定オブジェクトにセットする。このとき



必要ならばデータ型の変換が行われる。たとえば、タイマ設定ファイル内の時刻設定は、時刻設定をあらわす文字列として保持されている。これをそのままタイマ設定オブジェクトで利用するのは効率的ではない。そこで、時刻設定を整数値（たとえば 0時0分からの経過分）に変換することでより効率良く処理を行なうことができる。以下、各設定項目を設定オブジェクト側に設定するときは適当なデータ変換が行われると仮定する。

## 【0 1 7 9】

つぎに、タイマ設定ファイルの内容から関連する設定ファイルのリストをとりだし、それぞれから設定オブジェクトを作成する(S9a04、図11)。初期状態では、アップロード・タイマ設定タイプのタイマ設定オブジェクトだけが作成される。このため、ここで作成される設定オブジェクトはアップロード設定オブジェクトである。アップロード設定オブジェクトの生成および初期化処理はS9b01以下で実行される。

## 【0 1 8 0】

最後に、S9a04で作成された各設定オブジェクトのオブジェクトIDを、タイマ設定オブジェクトに設定する(S9a05)。

## 【0 1 8 1】

S9a01からS9a05までの処理はハードディスク(210)内の設定ファイル群(211)に含まれるすべてのタイマ設定ファイルを処理するまで繰り返される。

## 【0 1 8 2】

次に図11に示されるアップロード設定オブジェクトの生成・初期化処理について説明する。

## 【0 1 8 3】

タイマ設定のときと同様にアップロード設定ファイルの内容をメモリに読みだす(S9b01)。それからアップロード設定オブジェクト領域と映像データ取得設定オブジェクト領域を作成する(S9b02)。本実施例では、アップロード設定オブジェクトと映像データ取得設定オブジェクトを1つの設定ファイルから作成する。

## 【0 1 8 4】

メインメモリ上のアップロード設定ファイルの内容から各項目の値を映像デー

タ取得設定オブジェクト内にセットする(S9b03)。具体的には、映像パラメータ、ローカル映像パラメータの設定値を取り出して、その値を映像データ取得設定オブジェクトにセットする。(アップロード設定オブジェクトは他の設定ファイルのオブジェクトIDのみを保持するオブジェクトである。他の属性値は持たない。) 【0185】 それから、アップロード設定ファイルの内容からサーバ設定ファイルのファイル名をとりだし、このファイルからサーバ設定オブジェクトを作成・初期化する(S9b04、図13)。このサーバ設定オブジェクトは映像データの取得先であるカメラサーバに関する情報を保持している。その作成・初期化処理は、図13の処理で実行される。そして、作成されたサーバ設定オブジェクトのオブジェクトIDを、映像データ取得設定オブジェクト内にセットする(S9b05)。

## 【0186】

つぎに、アップロード設定ファイルの内容からデータ送信設定ファイルのリストをとりだし、それぞれからデータ送信設定オブジェクトを作成する(S9b06、図11)。この部分は図11のデータ送信設定オブジェクトの作成・初期化処理を実行することになる。

## 【0187】

最後に、S9b06で作成されたデータ送信設定オブジェクトのオブジェクトIDをアップロード設定オブジェクトにセットする。また、同時に初期化が終了した映像データ取得設定オブジェクトのオブジェクトIDもアップロード設定オブジェクト内にセットする(S9b07)。

## 【0188】

図11は、データ送信設定オブジェクトの生成・初期化処理をあらわしている。

## 【0189】

データ送信設定ファイルの内容をメモリ上に読みだし(S9c01)、データ送信設定オブジェクトを作成する(S9c02)。それから、メモリ上のデータ送信設定ファイルの内容から各属性値をデータ送信設定オブジェクト内にセットする(S9c03)。具体的には送信パラメータ、映像加工設定の値を設定する。

## 【0190】

次に、データ送信先をあらわすサーバ設定ファイルからサーバ設定オブジェクトを生成する(S9c04、図13)。ここでは図13の処理が実行される。また、映像情報データ設定ファイル名のリストから、映像情報データ設定オブジェクトを作成する(S9c05、図12)。最後に、上記サーバ設定オブジェクトのIDと映像情報データ設定オブジェクトのオブジェクトIDをデータ送信設定オブジェクトに登録すれば処理は終了である(S9c06)。

## 【0191】

図12は、映像情報データ設定オブジェクトの作成・初期化処理である。

## 【0192】

まず、映像情報データ設定ファイルの内容をメモリ上に読みだす(S9d01)。それから映像情報データ設定オブジェクトの領域を確保する(S9d02)。最後にメモリ上の映像情報データ設定ファイルの内容から各属性値を取り出し、映像情報データ設定オブジェクト内にセットする(S9c03)。ここで設定される属性値はテンプレート・ファイル名、データ作成パラメータの値である。

## 【0193】

図13は、S9b04又はS9c04における、カメラサーバ用、または送信先サーバ用の、サーバ設定オブジェクトの作成・初期化処理をあらわしている。

## 【0194】

他の設定オブジェクトとおなじく、サーバ設定ファイルの内容をメモリ上に読みだし(S9e01)、サーバ設定オブジェクトを作成する(S9e02)。次にメモリ上の内容から各属性値をサーバ設定オブジェクトに設定する(S9e03)。具体的にはホスト名、ポート番号、プロトコル名、接続パラメータの値を設定する。

## 【0195】

S9e04では経路設定ファイル名から経路設定オブジェクトを生成・初期化する。ここでは図14の処理を実行する。最後に、生成した経路設定オブジェクトのオブジェクトIDをサーバ設定オブジェクト内にセットする(S9e05)。

## 【0196】

図14は、S9e04における経路設定オブジェクトの作成・初期化処理である。

## 【0197】

経路設定ファイルの内容をメモリ上に読みだし(S9f01)、経路設定オブジェクトを作成する(S9f02)。次にメモリ上の内容から各属性値を経路設定オブジェクトに設定する。(S9f03)具体的にはモデムポート、電話番号。モデム接続パラメータ、ゲートウェイ・ホスト、ゲートウェイ・ポート番号、プロトコル名の値を設定する。

## 【 0 1 9 8 】

図15ではタイマ処理モジュール(103)によるタイマ処理の流れを説明する。

## 【 0 1 9 9 】

以下の処理は、タイマ設定オブジェクトのタイプ(アップロードタイマ設定タイプ、データ送信タイマ設定タイプ、ファイル有効期限タイマ設定タイプ)とは関係なく共通の処理である。以下の処理では、アップロードタイマ設定タイプのタイマ設定オブジェクトに対する処理を説明する。当然この処理はアップロードタイマ管理サブモジュール(104)内で実行される。

## 【 0 2 0 0 】

基本的な処理のながれはS10a01からS10a05までの短いフローチャートにまとめることができる。まず、タイマ設定オブジェクト内の時点設定(すなわち日付設定/曜日設定/時刻設定)の情報から、現在の時点を基準としてもっとも近い未来での「指定時点」を計算する(S10a01、図16)。この最も近い未来の指定時点のことを「次回処理時点」と呼ぶ。たとえば、時点設定が「午前0時0分から30分ごと」と指定されているとする。現在の時点(時刻)が午後2時10分とすると、次回処理時刻は午後2時30分となる。次回処理時刻の計算アルゴリズムについては後述する。

## 【 0 2 0 1 】

次回処理時点が計算の結果を判定して(S10a02)、エラーでなければ、次回処理時点になるまでこのタイマ処理はスリープする(S10a03)。アップロードタイマ管理サブモジュール(104)内に他にも処理すべきタイマ設定オブジェクトがあれば、スリープしている間にそちらを実行することも可能である。S10a02において、次回処理時点の計算がエラーとなったときはタイマ処理はそのまま終了する(S10a02)。

## 【0 2 0 2】

スリープが終了して次回処理時点になったら、タイマ設定オブジェクト内にセットされているすべてのアップロード設定オブジェクトに対してアップロード処理を実行させる(S10a04、図19)。具体的にはアップロード処理モジュール(107)に実行すべきアップロード設定オブジェクトを渡して、アップロード処理を開始してやればよい。この処理の詳細については、後で図19を用いて説明する。アップロード処理の実行は、タイマ処理と並列に実行することもできるし、アップロード処理の実行が終了するまではタイマ処理は停止状態になるようにしてもよい。本実施例ではアップロード処理が終了するまでタイマ処理は停止状態になるものとして説明をつづける。

## 【0 2 0 3】

アップロード処理が終了すると、タイマ処理側では映像アップロードサーバ(101)の終了処理が行われたか否かをチェックする(S10a05)。このチェックのためには、メインメモリ上に映像アップロードサーバ(101)の終了処理のフラグを用意しておく。そのフラグの値をしらべることでチェックする。

## 【0 2 0 4】

終了処理が行われたならば(S10a05)、タイマ処理は処理を終了する。終了処理が行われていなかったならば(S10a05)、次回処理時点の計算(S10a01)から処理を再開する。

## 【0 2 0 5】

次に図16を用いて次回処理時点の計算処理を説明する。 【0 2 0 6】

タイマ設定オブジェクト内の時点設定は、日付設定、曜日設定、時刻設定の3つの項目から成り立っているので、次回処理時点の計算処理は以下のように行っている。

## 【0 2 0 7】

まず、現在時点から、時刻設定をもとに次回処理時点の候補となる時点(候補時点)を計算する(S10b01、図17)。この処理の詳細は図17で説明する。計算に成功し、候補時点が得られたならば(S10b02)、S10b03以下の処理をおこなう。候補時点が得られなかった場合は(S10b02)、処理を終了しエラーを呼び出し先に返す

。例えば、時刻設定で、時間間隔が設定されていなければ、そのスケジュールによる次のジョブはないので、処理は終了する。

#### 【0 2 0 8】

候補時点を得られたならば、候補時点が曜日設定を満たすかをチェックする(S10b03)。その結果、候補時点が曜日設定を満足しない場合は(S10b04)、現在時点のかわりに候補時点を使って候補時点の再計算を行う(S10b01)。

#### 【0 2 0 9】

曜日設定を満足する候補時点を得られたならば、候補時点が日付設定を満たすかをチェックする(S10b05、図18)。この処理の詳細は図18で説明する。

#### 【0 2 1 0】

このチェックでは「満たす」、「満たさない」、「エラー」の3種類の結果が得られる。結果が「エラー」ならば(S10b06)、候補時点を再計算しても日付設定を満たす事ができないことをあらわしている。よって「エラー」の場合は処理を中止し、呼び出し側(S10a02)にエラーを返す。結果が「満たす」の場合は(S10b07)、得られた候補時点を次回処理時点の値として呼び出し側(S10a02)に返す。日付設定を「満たさない」場合は、現在時点のかわりに候補時点をつかってS10b01からS10b07までの処理を繰り返すことになる。

#### 【0 2 1 1】

図17では時刻設定からの候補時点の算出処理をあらわしたフローチャートである。

#### 【0 2 1 2】

図17において、現在時点ないし再計算の対象となっている候補時点にxという名前がついているとする。(実装上はxは変数名に相当する。) xの実体は "1999年2月1日1時10分" といった文字列でもよいし、西暦1年1月1日0時0分からの経過分で表現してもよい。本実施例は一般的な手法にしたがい、1970年1月1日0時0分からの経過分で候補時点をあつかうことにする。 【0 2 1 3】

まず、xから時刻を表わす部分を取り出し、H(x)と名前をつける(S10c01)。これは、

$H(x) = x \bmod (24 \times 60)$  という計算で得ることができる。(A mode BはAをB

で割った余りを計算する演算である。また、 $24 \times 60$ は一日 24 時間を分であらわした値である。) 【0 2 1 4】

もし、 $H(x)$ が開始時刻よりも小さな値のときは(S10c02)、開始時刻に $H(x')$ と名前を付ける(S10c03)。 $H(x)$ が開始時刻より大きな値のときは(S10c02)、つぎに $H(x)$ と終了時刻を比較する。

【0 2 1 5】

$H(x)$ が終了時刻より大きな値の場合は(S10c04)、(開始時刻 +  $24 \times 60$ )を $H(x')$ とする(S10c05)。これは翌日の開始時間に相当する。

【0 2 1 6】

$H(x)$ が終了時間より小さい場合は(S10c04)、開始時刻から時刻間隔 $\times N$ ( $N$ は0以上の整数)経過した時刻で、 $H(x)$ の直後になる時刻を $H(x')$ とする(S10c06)。このときは

$$H(x') = \{ (H(x) \text{ div } \text{時刻間隔}) + 1 \} * \text{時刻間隔}$$

で値をもとめることができる。(ただし divは整数演算での割り算である。したがって  $3 \text{ div } 2$ の値は1.5ではなく、1となる。) 【0 2 1 7】

最後に、候補時点 $x'$ を以下の式で計算する(S10c07)。

【0 2 1 8】

$$x' = (x - H(x)) + H(x') \quad \text{【0 2 1 9】}$$

曜日設定のチェックはまず、候補時点の曜日を計算する。曜日の計算はZellerの公式等を用いて簡単に計算することができる。曜日設定は他の設定とはちがい、設定された曜日のリストを保持している。候補時点から計算された曜日がリスト内に存在するか否かをチェックし、その結果を呼び出し側に返せばよい。

【0 2 2 0】

図18は、S10b05における日付設定に基づくチェック処理を示したものである。日付設定には時刻設定と同様に開始日付、終了日付、日付間隔が設定されている。したがって、チェックの方式は時刻設定から候補時点を計算するやりかたと似た手順となっている。

【0 2 2 1】

まず、候補時点 $x$ から日付をあらわす部分を取り出す(S10d01)。これを $D(x)$ と

する。D(x)の値は

$D(x) = x \text{ div } 24 \times 60$ で計算することができる。

【0 2 2 2】

D(x)が開始日付より小さい場合は(S10d02)、返り値として「満たさない」を返し処理を終了する(S10d05)。D(x)が終了日付より大きい場合は(S10d06)、返り値として「エラー」を返し処理を終了する(S10d11)。

【0 2 2 3】

どちらでもない場合は、開始日付から日付間隔\*N(Nは0以上の整数)経過した日付のなかで、D(x)と一値する日付がないか調べる(S10d07)。

【0 2 2 4】

一致する日付がみつかった場合は「満たす」を返す(S10d08、 S10d09)。見つからなかった場合は、「満たさない」となる(S10d08、 S10d10)。

【0 2 2 5】

図19では、アップロード処理について説明する。

【0 2 2 6】

タイマ処理モジュール(103)やコマンド実行サブモジュール(127)からアップロード処理モジュール(107)が呼び出されると、呼び出し時に指定されたアップロード設定オブジェクトの内容をもとに以下の処理を実行する。

【0 2 2 7】

全体的な処理の流れは以下の通りである。まず映像データをカメラサーバから取得し(S11a01、図20)、映像データを加工する(S11a02)。この加工処理は、外部コマンド(プログラム)で実行される。それから映像データをWWWホストに送信する(S11a03、図21)。S11a02からS11a03までの処理は、アップロード設定オブジェクトから参照されている全てのデータ送信設定オブジェクトに対して実行される(S11a04)。

【0 2 2 8】

図20は、S11a01の映像データ取得処理を細分化したものである。

【0 2 2 9】

まず、映像取得設定オブジェクトから参照しているサーバ設定オブジェクトを



取り出して、サーバ接続処理を行う (S11c01、図23)。なお、このサーバ設定オブジェクトにはカメラサーバに関する情報が設定されている。そこでサーバ設定管理モジュールには、カメラサーバ接続処理を依頼する。

## 【 0 2 3 0 】

サーバ接続処理が実行されたならば、次に映像パラメータをサーバ設定管理モジュールに渡してカメラサーバからの映像取得を依頼する (S11b02)。この依頼は、サーバ設定管理モジュール内のwvtcpプロトコル処理サブモジュールによって処理され、カメラサーバが処理する事のできるメッセージ形式(すなわちwvtcpプロトコルの映像取得メッセージ)に変換されて、カメラサーバに送信される。

## 【 0 2 3 1 】

カメラサーバは要求に応じて映像データをwvtcpプロトコル処理サブモジュールに送信する。また、送信されてきた映像パラメータも返送する。このとき、要求された映像パラメータの値を満たすことができなかった場合は、映像パラメータの値を実際の値に変更してから送り返す。たとえば、映像表示サイズの値を320x240ドットと指定したが、カメラサーバ側の事情により240x180ドットの映像データが送信された場合、返送されてきた映像パラメータ内の映像表示サイズの値は240x180に変更されることになる。カメラサーバの種類によっては、変更されたパラメータを返送する機能がないものも考えられる。この場合、サーバ通信処理モジュール(117)はパラメータに変更がなかったものとして処理をおこなう。

## 【 0 2 3 2 】

受け取られた映像データは一時的にサーバ通信処理モジュール(117)によって保持される。この時点で映像パラメータ送信処理(S11b02)が終了するので制御は映像データ取得サブモジュール(110)にもどる。映像制御サブモジュールでは映像データおよび映像パラメータをサーバ通信処理プログラムから受け取る (S11b03)。映像データと映像パラメータ内の情報を映像情報データベース(116)に登録する (S11b04、図22)。

## 【 0 2 3 3 】

図21は、S11a04におけるデータ送信処理の内容を示すものである。

## 【 0 2 3 4 】

まず、データ送信設定オブジェクトから参照先のサーバ設定オブジェクトを取り出し、サーバ接続処理を実行する(S11c01)。なお、このサーバ設定オブジェクトには送信先サーバに関する情報が設定されている。

【0 2 3 5】

サーバとの接続処理が完了したら、映像情報データベース(116)から映像データを取得する(S11c02)。(ここで、外部コマンドで実行された映像加工処理は加工した映像データを映像情報データベース(116)に登録するものとする。)

【0 2 3 6】

データ送信設定から送信パラメータを取得し、映像データをいっしょにサーバ通信処理モジュール(117)に渡してWWWホストへのデータ送信を依頼する(S11c03)。サーバ通信処理モジュール(117)側では、ftpプロトコル処理モジュールやhttpプロトコル処理モジュールを経由して映像データをWWWホストに送信する。

【0 2 3 7】

送信処理が終了すると、映像情報データアップロード処理(すなわち作成と送信処理)を実行する(S11c04、図22)。この処理については図22で説明する。この処理はデータ送信設定から参照できるすべての映像情報データ設定オブジェクトを処理するまでおこなう(S11c05)。

【0 2 3 8】

図22は映像情報データのアップロード処理をあらわすフローチャートである。

【0 2 3 9】

まず、映像情報データ設定オブジェクト内のテンプレートファイルからテンプレートデータをメモリに読み込む(S11d01)。

【0 2 4 0】

このテンプレートデータの先頭から単語を取り出し(S11d02)、単語の両端が”@”で囲まれているかチェックする(S11d03)。囲まれている場合は(S11d03)キーワードではないので、そのまま情報データ作成用のメモリ領域に書き込む。囲まれている場合は(S11d03)、映像情報データベース(116)内をキーワードをキーとして値を検索する(S11d04)。値が検索できた場合は(S11d05)、その値を映像情報データ作成用のメモリ領域に書き込む(S11d06)。値がみつからなかった場合は(S

11d05)キーワードをそのまま情報データ作成用のメモリ領域に書き込む。S11d02からS11d06までの処理はテンプレートデータ内のすべての単語をチェックするまで繰り返される(S11d07)。

【 0 2 4 1 】

最後に、データ送信サブモジュール(112)を使って作成した映像情報データをWWWホストに送信する(S11d08)。この処理の詳細は映像データの送信処理とまったく同じである。

【 0 2 4 2 】

図23は、S11b01におけるサーバ通信処理モジュール(117)で実行されるサーバ接続処理のフローチャートである。なお、図23の処理は、カメラサーバーへの接続と、WWWサーバへの接続を兼ねて説明する。

【 0 2 4 3 】

サーバ接続処理が呼び出される時には、かならず参照すべきサーバ設定オブジェクトが指定される。まず、このサーバ設定オブジェクトが参照している経路情報オブジェクトからモデム関連の属性値(モデムポート/電話番号/モデム接続パラメータ)を取り出す(S11e01)。

【 0 2 4 4 】

モデム関連の属性値が設定されているときは(S11e02)、その設定を用いてモデムによる接続処理を行う(S11e03)。この処理はOSによって処理されるので、必要なパラメータを指定してOSインタフェースモジュール(102)を呼び出せばよい。

【 0 2 4 5 】

次に、経路情報オブジェクトからゲートウェイに関する設定値を取得する(S11e04)。設定されているならば(S11e05)、ゲートウェイとの接続処理を行う(S11e06)。この処理もOSインタフェースを利用して実施される。最後に接続先のゲートウェイにサーバに関する情報(ホスト名、ポート番号、プロトコル名)を送信し、送信メッセージの中継処理を依頼する(S11e08)。以降は、サーバと直接接続を行う場合とまったく同じやり方でメッセージのやり取りが可能となる。

【 0 2 4 6 】

また、ゲートウェイの情報が設定されていなかった場合は(S11e05)、サーバ設

定情報に設定されているサーバとの接続処理を行う(S11e07)。この処理はソケットなどのプロセス間通信機構をOSインタフェースを介して利用することで実施される。

#### 【0 2 4 7】

上記の本実施例によれば、あらかじめ設定した時点に適切な処理を開始するタイム処理を行う、つまり、予め設定したスケジュールに応じて、カメラを制御して、映像を取得したり、所定のサーバーに映像データを転送したりすることが出来るので、自動的なアップロードサービスを提供することが出来る。

#### 【0 2 4 8】

また、スケジュールに従って取得した映像情報を、メモリーやハードディスクなどの格納手段に複数枚格納しておくことによって、過去の映像も配信することが出来る。

#### 【0 2 4 9】

また、カメラで取得した映像データを加工編集して、インターネットへ向けて配信するのに適した形式のデータを生成するので、容易にインターネットを介して、映像データを配信することが出来る。

#### 【0 2 5 0】

また、テンプレート情報として、映像データの取得日時、映像データのサイズ、タイトル、カメラ制御情報のうち、少なくともひとつの情報を、前記映像データとともに表示させるための雛形を有するので、映像を取得してみても分かる情報でも、手間をかけることなく、映像データとともにユーザーに提供することが出来る。

#### 【0 2 5 1】

また、使用者の加工編集要求を受け付けて、前記映像データを加工編集するアプリケーションプログラムを制御することによって、使用者の好みに応じて、多彩な加工処理が可能となる。

#### 【0 2 5 2】

さらに、スケジュールの設定を、時間や日付や曜日での指定を出来るようにすることによって、スケジュール設定の自由度を非常に高くすることが出来る。

【 0 2 5 3 】

また、任意のサーバーを指定して、転送することも出来るので、カメラで撮影するだけのサーバーや、映像データを蓄積するだけのサーバーや、インターネットに映像を配信するためのサーバーといった具合に、機能を分割したシステムも容易に形成することが出来るし、映像配信サービス提供のための多様な運用形態にも、柔軟に応えることが出来る。

【 0 2 5 4 】

さらに、データ転送経路を管理することによって、たとえば、障害のある経路を避けて映像データを転送したり、モデム接続やゲートウェイ・ホスト接続を、任意に設定できる。

【 0 2 5 5 】

また、格納手段に格納された映像データの有効期限を管理し、有効期限の経過した映像を、前記格納手段から消去することによって、格納手段の容量が一杯になってしまって、映像を撮り込めなくなってしまうことを防止することが出来る。

【 0 2 5 6 】

また、テンプレート情報として、多くの種類を備えることによって、多様化する表示形態にも、柔軟に対応し、表現力豊かな映像配信サービスを提供することが出来る。

【 0 2 5 7 】

また、ネットワーク上のユーザーからの指示を受け付けて、カメラを制御して、任意のタイミングで映像データを取得することが出来るので、ユーザーの多様な、または急な要求にも応えることが出来る。

【 0 2 5 8 】

なお、本実施例においては、映像データの有効期限を管理して、この期限が過ぎたものを消去するようにしていたが、期限が来た映像から上書きを許可するようにしてもよいし、期限が来ていて、かつ、記憶手段の残容量が少なくなった場合に、古い映像から消去、または上書きをするようにしてもよい。または、着脱式の記憶手段を用いるような装置であれば、新たな記憶手段に交換するようにし

てもよい。なお、キーワードに対応する情報は、映像情報データベース（116）に記憶させていたが、例えば映像ファイルとは別のファイルに記憶させてもいいし、画像ファイルのヘッダー部等でもいい。

#### 【0259】

請求の範囲に記載の、「映像データとは別に」とは、少なくとも、映像データにうめこまなければ、いいことを示す。

#### 【0260】

#### 〔第2実施例〕

図24は第2実施例の構成図である。第1実施例との相違点のみ説明する。図24では、映像アップロードサーバ(101)に相当する機能が複数のサーバ・プログラムに分割されている。それぞれ、同一あるいはネットワークで結合されたコンピュータ上で稼働する。

#### 【0261】

これらのコンピュータのハードウェア構成は図2でしめした映像アップロードホスト(201)と同じものである。タイマ管理サーバ(1201)はタイマ処理モジュール(103)に相当する機能を提供するサーバ・プログラムである。映像取得サーバ(1202)は映像データ取得サブモジュール(110)に相当するサーバ・プログラムである。映像加工サーバ(1203)は映像データ加工サブモジュール(111)に相当するサーバ・プログラムである。映像送信サーバ(1205)はデータ送信サブモジュール(112)/テンプレート・データ取得サブモジュール(113)/情報データ作成サブモジュール(G14)/データ送信サブモジュール(112)に相当するサーバ・プログラムである。

#### 【0262】

映像情報データベース(116)に相当する機能はデータベース管理システム(1205)が担っている。また、サーバ通信処理モジュール(117)はすべてのサーバに組み込まれているものとする。各サーバは117を使って互いにデータを交換する。ただし、ftpプロトコル処理サブモジュール(120)、httpプロトコル処理サブモジュール(121)、wvtcpプロトコル処理モジュール(123)に相当する機能は、ftp中継サーバ(1207)/http中継サーバ(1206)/カメラサーバ中継サーバ(1208)として独立し

たサーバとなっている。各サーバの処理内容および設定データの内容は実施例1とほとんど同じである。ただし、モジュール間のやり取りはサーバ通信処理モジュールを介したプロセス間通信の形で実施される。

【 0 2 6 3 】

本実施例では、各モジュールを独立したサーバにしたことで負荷分散が用意となる。例えば、比較的処理が複雑な映像加工サーバや映像送信サーバは高速なCPUをもつホストで動作させ、他のサーバは低速だが安価なホストで稼働させる事が可能である。

【 0 2 6 4 】

また、あたらしいタイプのカメラサーバから映像を取得するための変更を考えると、実施例2には固有の利点がある。すなわち、実施例1ではプロトコル処理サブモジュールを追加するために映像アップロードサーバ(101)を作り直す必要がある。これに対し、第2実施例では新しいプロトコル処理用のサーバを作成し稼働させるだけでよい。

【 0 2 6 5 】

なお、本発明は複数の機器（たとえばホストコンピュータ、インタフェース機器、リーダ、プリンタ等）から構成されるシステムに適用しても、または一つの機器（たとえば複写機、ファクシミリ装置）からなる装置に適用してもよい。

【 0 2 6 6 】

また前述した実施形態の機能を実現する様に各種のデバイスを動作させる様に該各種デバイスと接続された装置あるいはシステム内のコンピュータに、前記実施形態機能を実現するためのソフトウェアのプログラムコードを供給し、そのシステムあるいは装置のコンピュータ（CPUあるいはMPU）を格納されたプログラムに従って前記各種デバイスを動作させることによって実施したものも本願発明の範疇に含まれる。

【 0 2 6 7 】

またこの場合、前記ソフトウェアのプログラムコード自体が前述した実施形態の機能を実現することになり、そのプログラムコード自体、及びそのプログラムコードをコンピュータに供給するための手段、例えばかかるプログラムコードを

格納した記憶媒体は本発明を構成する。

【0268】

かかるプログラムコードを格納する記憶媒体としては例えばフロッピーディスク、ハードディスク、光ディスク、光磁気ディスク、CD-ROM、磁気テープ、不揮発性のメモリカード、ROM等を用いることが出来る。

【0269】

またコンピュータが、供給されたプログラムコードを実行することにより、前述の実施形態の機能が実現されるだけではなく、そのプログラムコードが、コンピュータにおいて稼働しているOS(オペレーティングシステム)、あるいは他のアプリケーションソフト等と共同して前述の実施形態の機能が実現される場合にもかかるプログラムコードは本願発明の実施形態に含まれることは言うまでもない。

【0270】

更に供給されたプログラムコードが、コンピュータの機能拡張ボードやコンピュータに接続された機能拡張ユニットに備わるメモリに格納された後そのプログラムコードの指示に基づいてその機能拡張ボードや機能格納ユニットに備わるCPU等が実際の処理の一部または全部を行い、その処理によって前述した実施形態の機能が実現される場合も本願発明に含まれることは言うまでもない。

【0271】

【発明の効果】

以上、本発明によれば、あらかじめ設定した時点に適切な処理を開始するタイマ処理を行う、つまり、予め設定したスケジュールに応じて、カメラを制御して、映像を取得したり、所定のサーバーに映像データを転送したりすることが出来るので、自動的なアップロードサービスを提供することが出来る。

【0272】

また、スケジュールに従って取得した映像情報を、メモリーやハードディスクなどの格納手段に複数枚格納しておくことによって、過去の映像も配信することが出来る。

【0273】



また、カメラで取得した映像データを加工編集して、インターネットへ向けて配信するのに適した形式のデータを生成するので、容易にインターネットを介して、映像データを配信することが出来る。

【 0 2 7 4 】

また、テンプレート情報として、映像データの取得日時、映像データのサイズ、タイトル、カメラ制御情報のうち、少なくともひとつの情報を、前記映像データとともに表示させるための雛形を有するので、映像を取得して見て分かる情報でも、手間をかけることなく、一緒に映像に写し込むことなく、映像データとともにユーザーに提供することが出来る。

【 0 2 7 5 】

また、使用者の加工編集要求を受け付けて、前記映像データを加工編集するアプリケーションプログラムを制御することによって、使用者の好みに応じて、多彩な加工処理が可能となる。

【 0 2 7 6 】

さらに、スケジュールの設定を、時間や日付や曜日での指定を出来るようにすることによって、スケジュール設定の自由度を非常に高くすることが出来る。

【 0 2 7 7 】

また、任意のサーバーを指定して、転送することも出来るので、カメラで撮影するだけのサーバーや、映像データを蓄積するだけのサーバーや、インターネットに映像を配信するためのサーバーといった具合に、機能を分割したシステムも容易に形成することが出来るし、映像配信サービス提供のための多様な運用形態にも、柔軟に応えることが出来る。

【 0 2 7 8 】

さらに、データ転送経路を管理することによって、たとえば、障害のある経路を避けて映像データを転送したり、モデム接続やゲートウェイ・ホスト接続を、任意に設定できる。

【 0 2 7 9 】

また、格納手段に格納された映像データの有効期限を管理し、有効期限の経過した映像を、前記格納手段から消去することによって、格納手段の容量が一杯に

なってしまって、映像を撮り込めなくなってしまうことを防止することが出来る。

【0280】

また、テンプレート情報として、多くの種類を備えることによって、多様化する表示形態にも、柔軟に対応し、表現力豊かな映像配信サービスを提供することが出来る。

【0281】

また、ネットワーク上のユーザーからの指示を受け付けて、カメラを制御して、任意のタイミングで映像データを取得することが出来るので、ユーザーの多様な、または急な要求にも応えることが出来る。

【図面の簡単な説明】

【図1】

実施例1のソフトウェア構成をあらわす構成図

【図2】

実施例1のハードウェア構成をあらわす構成図

【図3】

実施例1のデータ構成をあらわす構成図

【図4】

従来例の概要図

【図5】

従来例のユーザインタフェース

【図6】

実施例1の概要図

【図7】

実施例1のユーザインタフェースの例をしめす図

【図8】

テンプレートファイルの例をしめす図

【図9】

タイマ設定オブジェクト作成のフローチャート

【図 1 0】

アップロード設定オブジェクト作成のフローチャート

【図 1 1】

データ送信設定オブジェクト作成のフローチャート

【図 1 2】

映像情報データ設定オブジェクト作成のフローチャート

【図 1 3】

送信先サーバ用又はカメラサーバ用設定オブジェクト作成のフローチャート

【図 1 4】

経路設定オブジェクト作成のフローチャート

【図 1 5】

タイマ処理のフローチャート

【図 1 6】

次回処理時点の計算のフローチャート

【図 1 7】

候補時点の算出のフローチャート

【図 1 8】

候補時点の日付設定のチェックのフローチャート

【図 1 9】

アップロード処理のフローチャート

【図 2 0】

映像データ取得処理のフローチャート

【図 2 1】

映像データ送信処理のフローチャート

【図 2 2】

映像データのアップロード処理のフローチャート

【図 2 3】

サーバ接続処理のフローチャート

【図 2 4】

## 第 2 実施例の概略構成図のフローチャート

### 【符号の説明】

- 1 0 1 映像アップロードサーバ
- 1 0 2 映像アップロードサーバ内の各モジュールと O S 間のインタフェースを提供する O S インタフェースモジュール
- 1 0 3 指定した時点に適当な処理を起動するタイマ処理モジュール
- 1 0 4 映像アップロード処理用のタイマを管理するアップロードタイマ管理サブモジュール
- 1 0 5 データ送信処理用のタイマを管理するデータ送信タイマ管理サブモジュール
- 1 0 6 データ削除処理用のタイマを管理するデータ有効期限タイマ管理サブモジュール
- 1 0 7 映像データのアップロード処理を制御するアップロード処理モジュール
- 1 1 0 映像データをカメラサーバから取得する映像データ取得サブモジュール
- 1 1 1 映像データを加工する映像データ加工サブモジュール
- 1 1 2 データを WWW ホストに送信するデータ送信サブモジュール
- 1 1 3 映像情報データの雛形であるテンプレートデータを取得するテンプレートデータ取得サブモジュール
- 1 1 4 映像情報データを作成する映像情報データ作成サブモジュール
- 1 1 5 有効期限がすぎたデータを削除するデータ削除処理サブモジュール
- 1 1 6 映像に関する情報を一時的に保管する映像情報データベース
- 1 1 7 カメラサーバ等のサーバとの通信を処理するサーバ通信処理モジュール
- 1 1 8 サーバとの接続処理をおこなうサーバ接続処理サブモジュール
- 1 1 9 モデムやゲートウェイ・ホストを介した接続処理をおこなう経路接続処理サブモジュール
- 1 2 0 f t p プロトコルによる通信を処理する f t p プロトコル処理サブモ

ジュール

1 2 1 h t t p プロトコルによる通信を処理する h t t p プロトコル処理サブモジュール

1 2 2 w v t c p プロトコルによる通信を処理する w v t c p プロトコル処理サブモジュール

1 2 5 ターミナルクライアントによるユーザの操作を処理するためのターミナル処理モジュール

1 2 6 ユーザが入力したコマンドを受信するためのコマンド受信サブモジュール

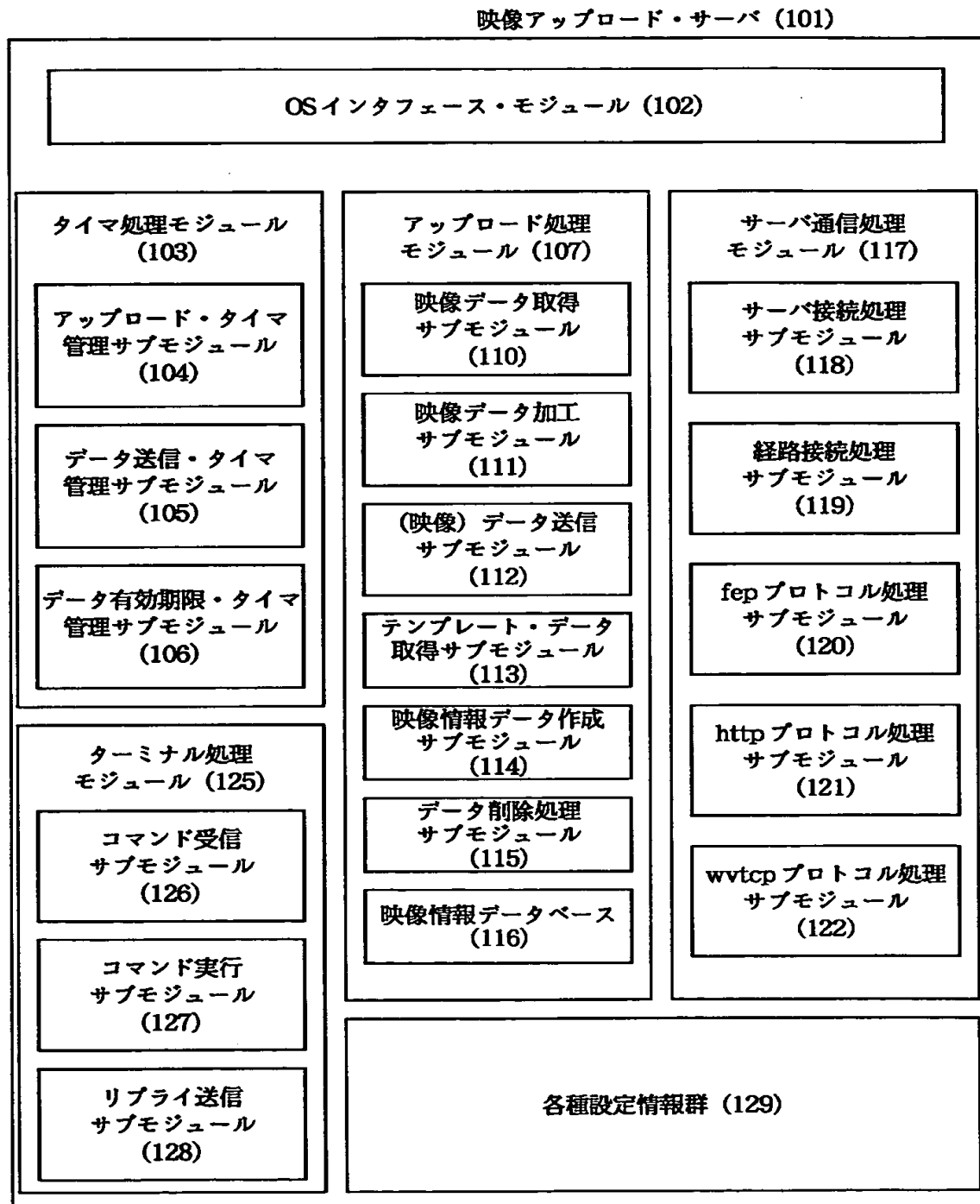
1 2 7 受け取ったコマンドを解釈し実行するコマンド実行サブモジュール

1 2 8 コマンド実行の結果をユーザに送り返すリプライ送信サブモジュール

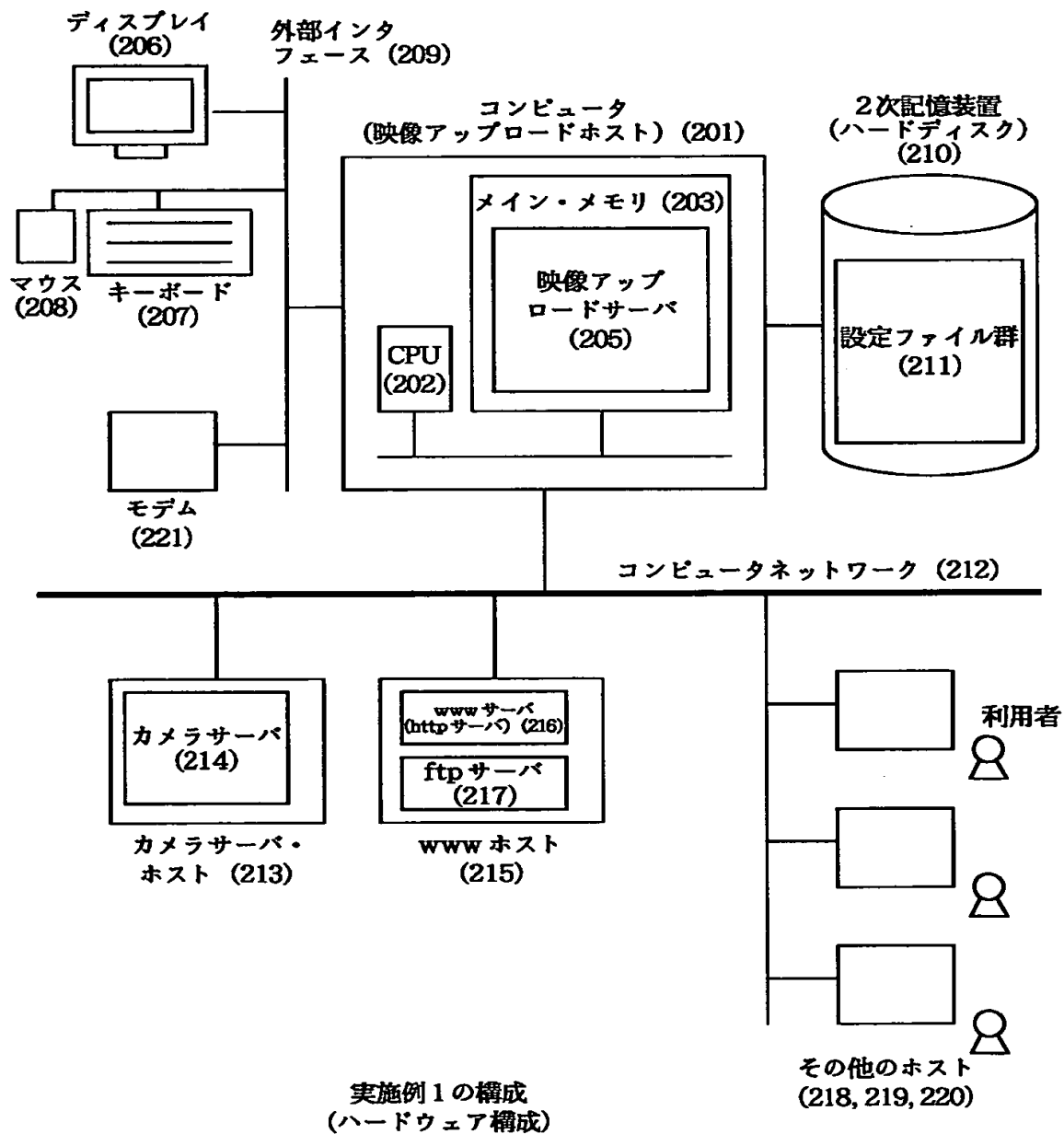
1 2 9 上記各モジュールで使用されるさまざまな設定情報のあつまりをあらわす各種設定情報群

【書類名】 図面

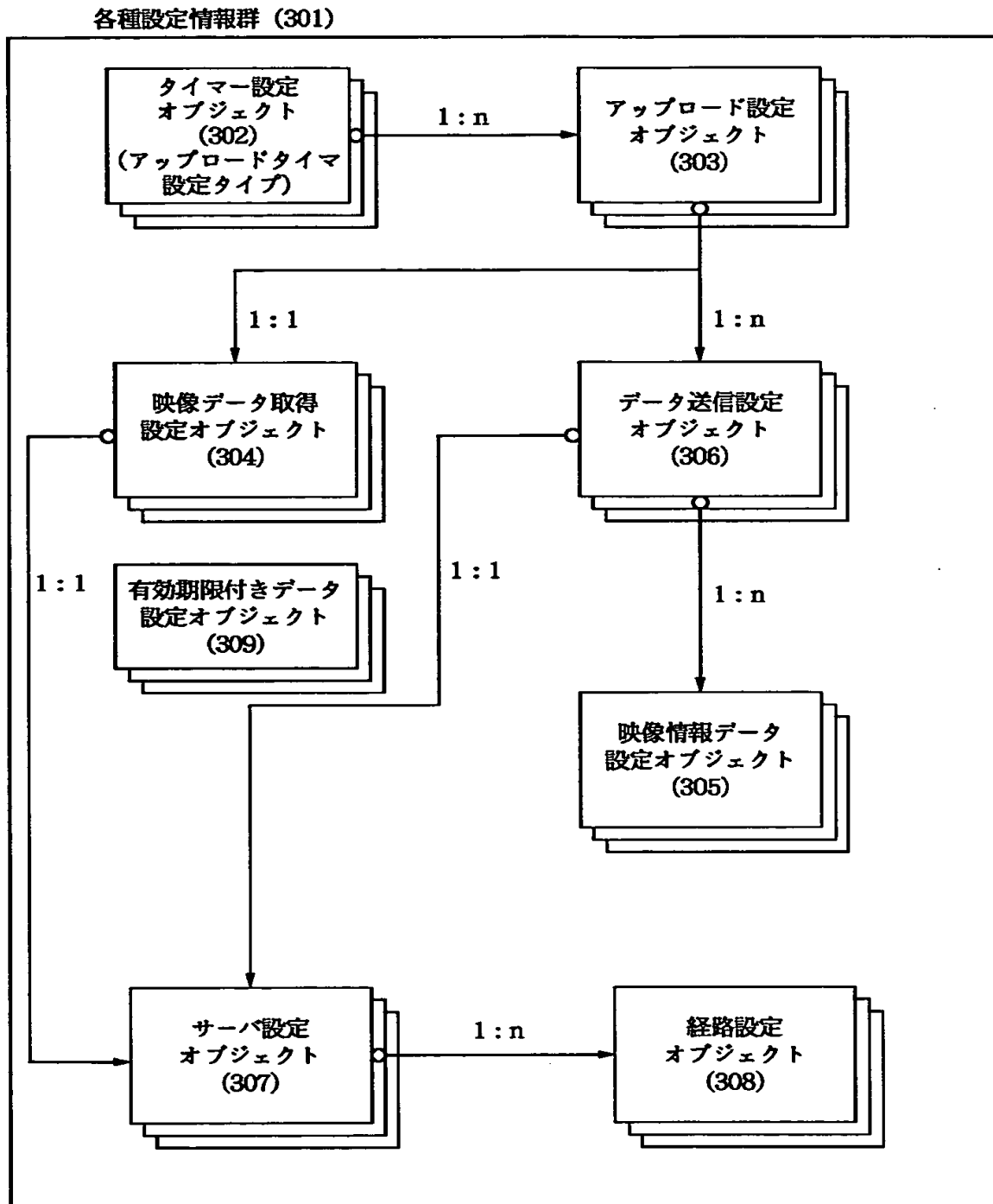
【図 1】



【図 2】



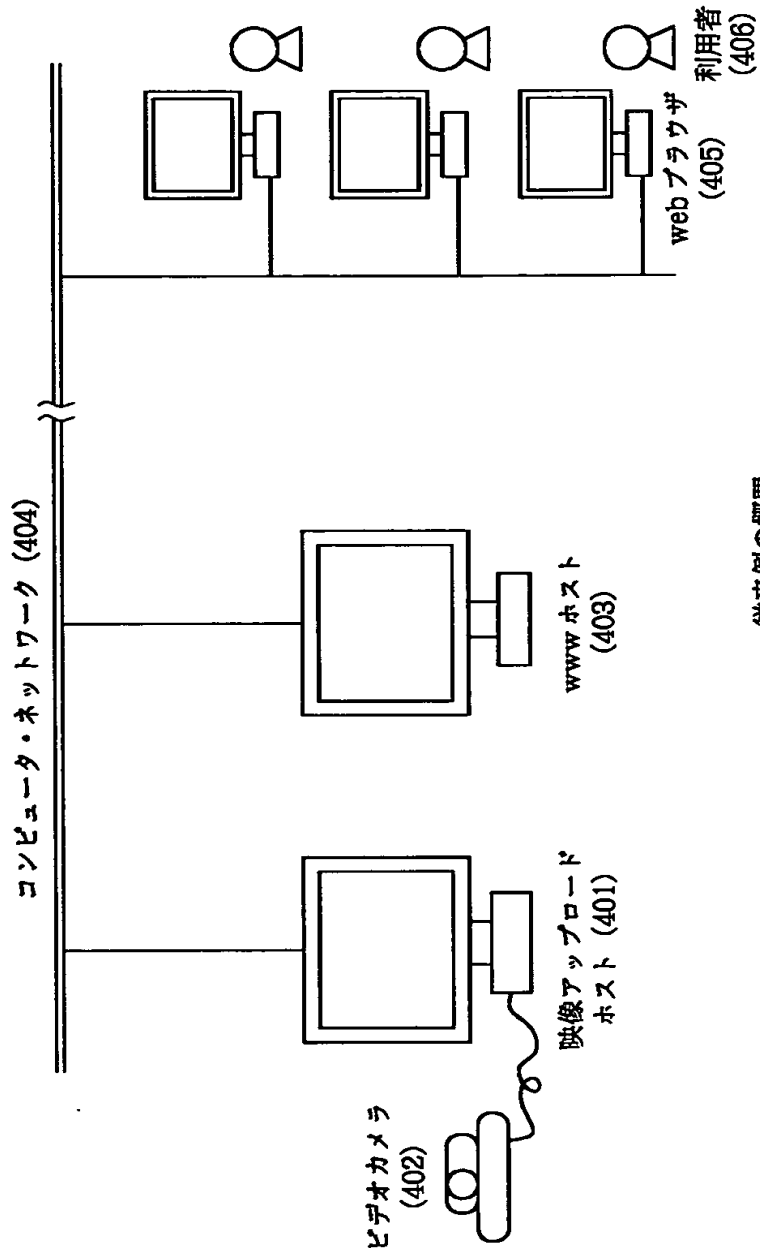
【図 3】



実施例 1 の構成  
(データ構成)

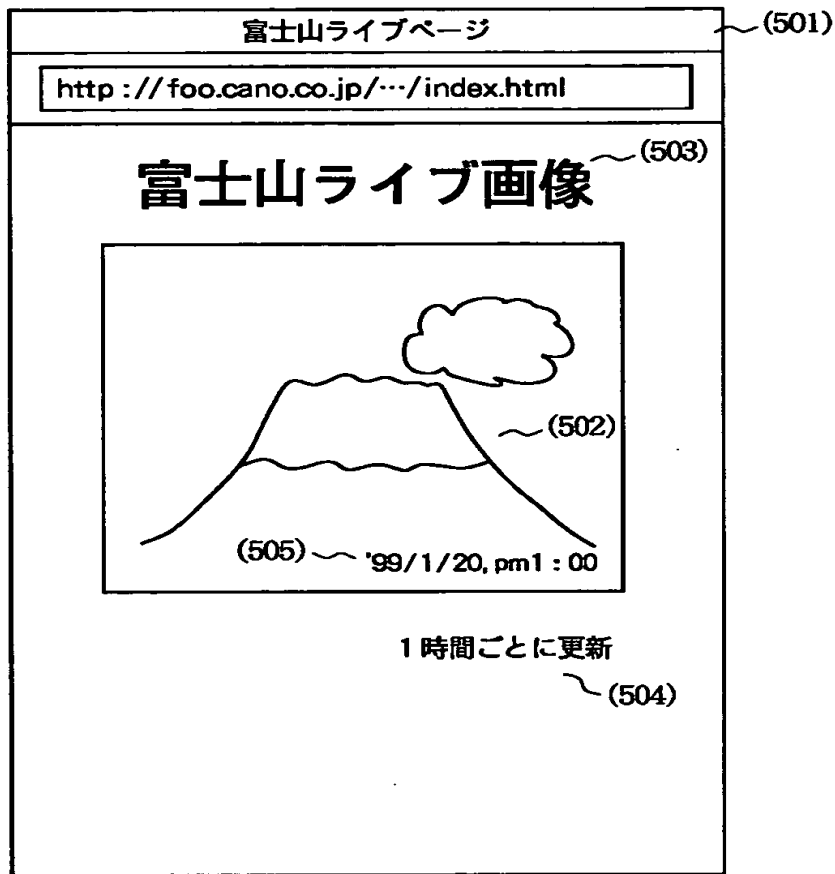


【図 4】



従来例の概要

【図 5】



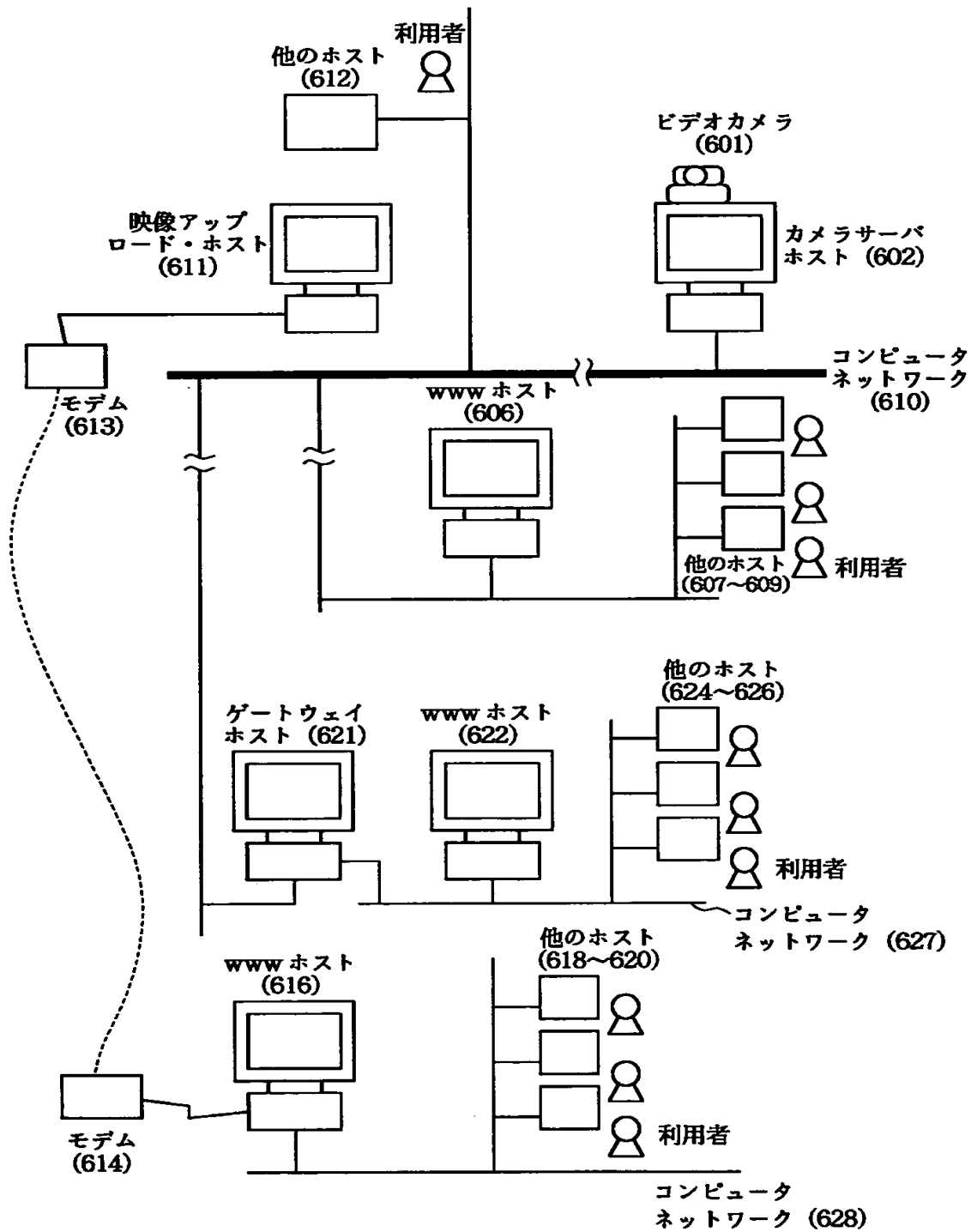
```

<html>
  <head>
    <title>富士山ライブ・ページ</title>
    <h1>富士山ライブ画像</h1>
  </head>
  <body>
    
    <p>
      1 時間ごとに更新
    </p>
  </body>
</html>

```

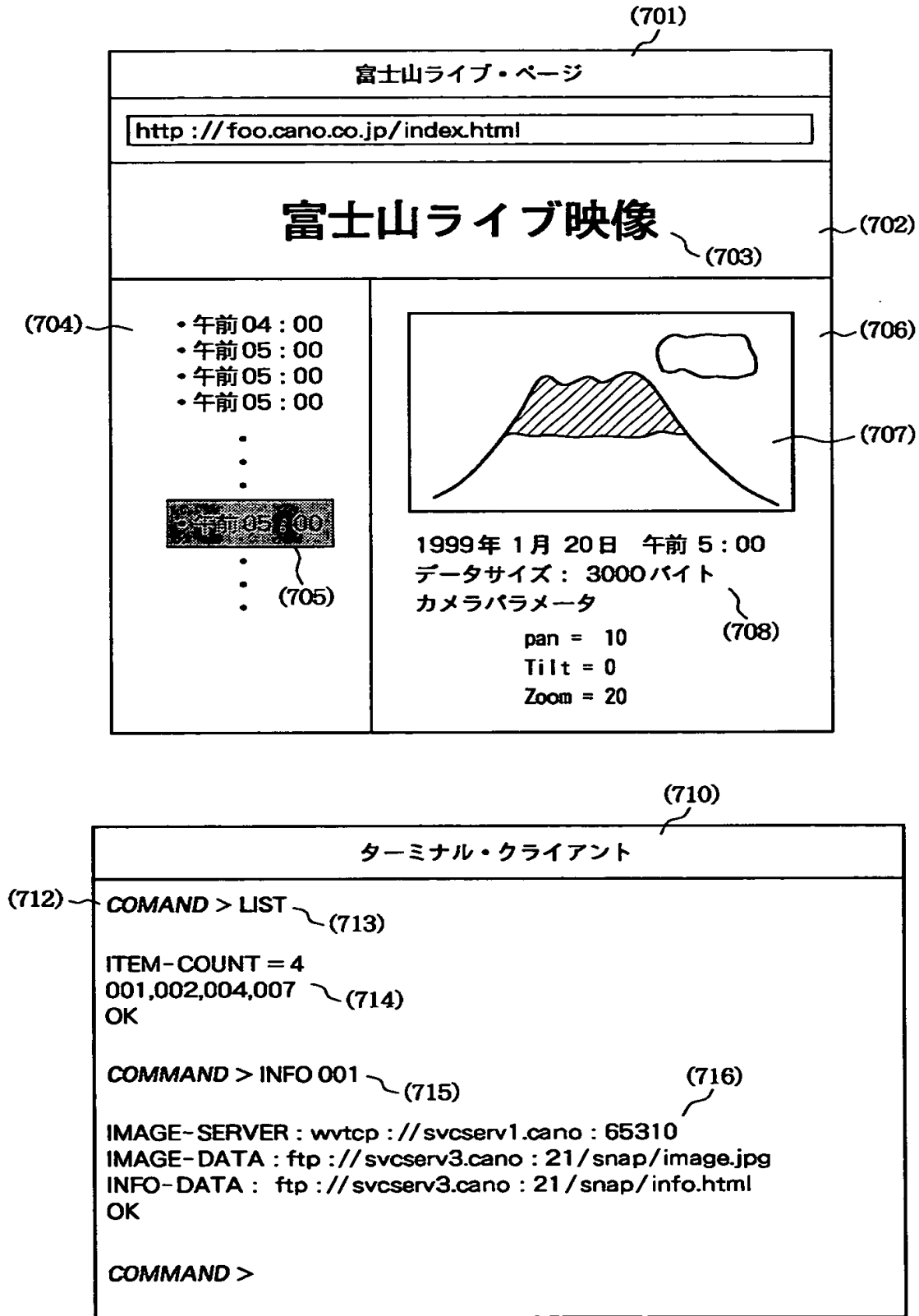
従来例のユーザインタフェース

【図 6】



実施例 1 の概要

【図 7】



実施例 1 のユーザインタフェース

【図 8】

```
< html >
  < head >
    < title >テンプレートファイルのサンプル</ title >
  </ head >

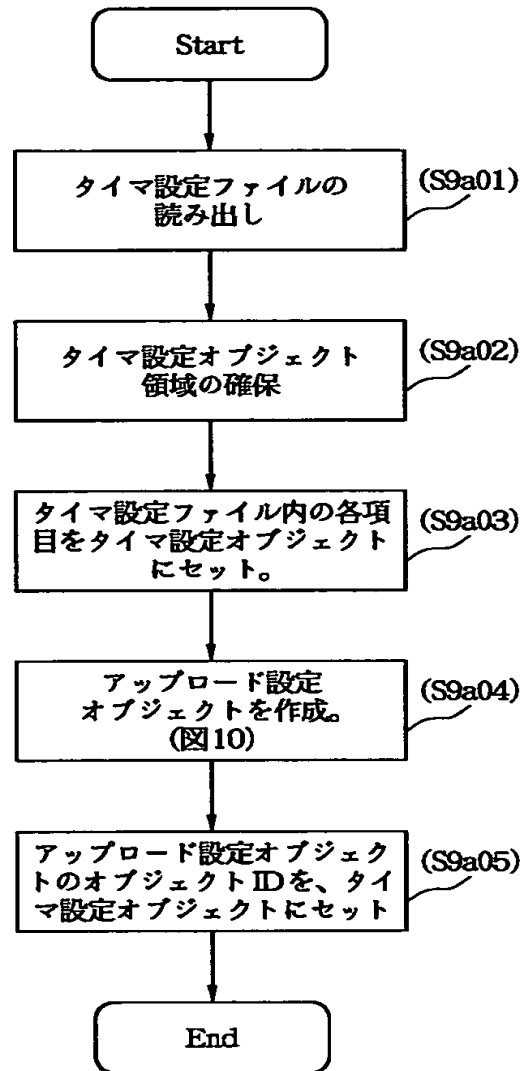
  < body >
    @@ IMAGE_CAPTION @@ < p >
    < p >
      日付 : @@ DATE_STAMP [% Y 年 % m 月 % d 日 (% a)] @@ < p >
      時刻 : @@ DATE_STAMP [% H : % M] @@ < p >
    < p >
      < image src = " @@ IMAGE_FILE @@ " >
    </ body >
  </ html >
```

```
< html >
  < head >
    < title >テンプレートファイルのサンプル</ title >
  </ head >

  < body >
    富士山の映像 < p >
    < p >
      日付 : 1999 年 1 月 21 日 ( 火 ) < p >
      時刻 : 午前 10 : 00 < p >
    < p >
      < image src = " mt_Fuji.jpg " >
    </ body >
  </ html >
```

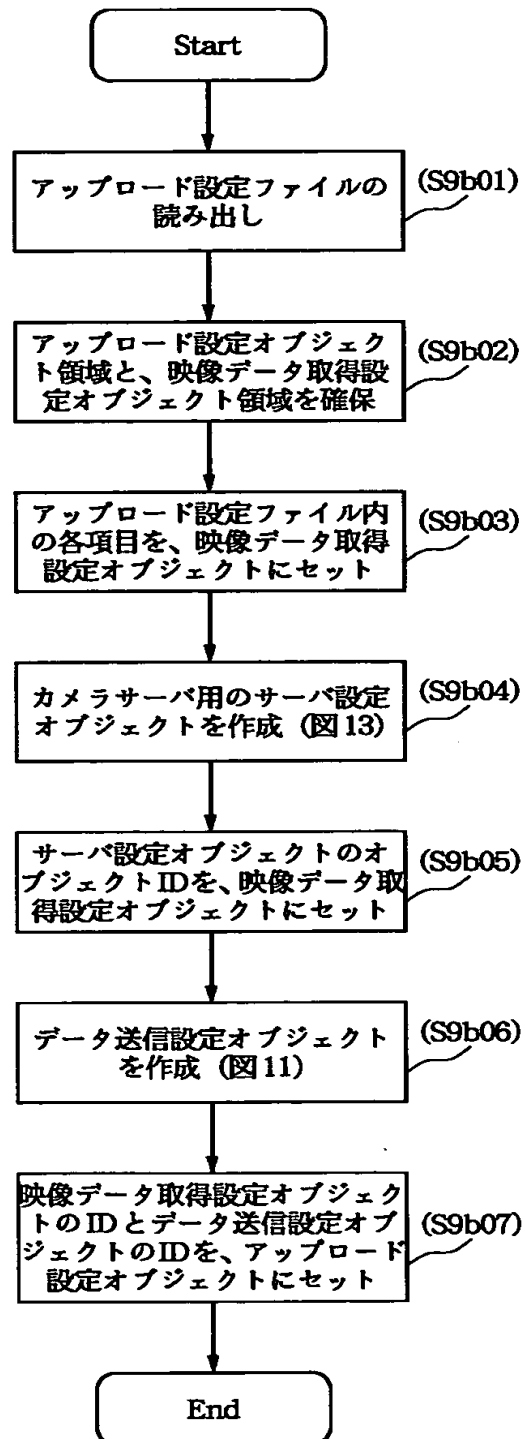
テンプレートファイルの例

【図 9】



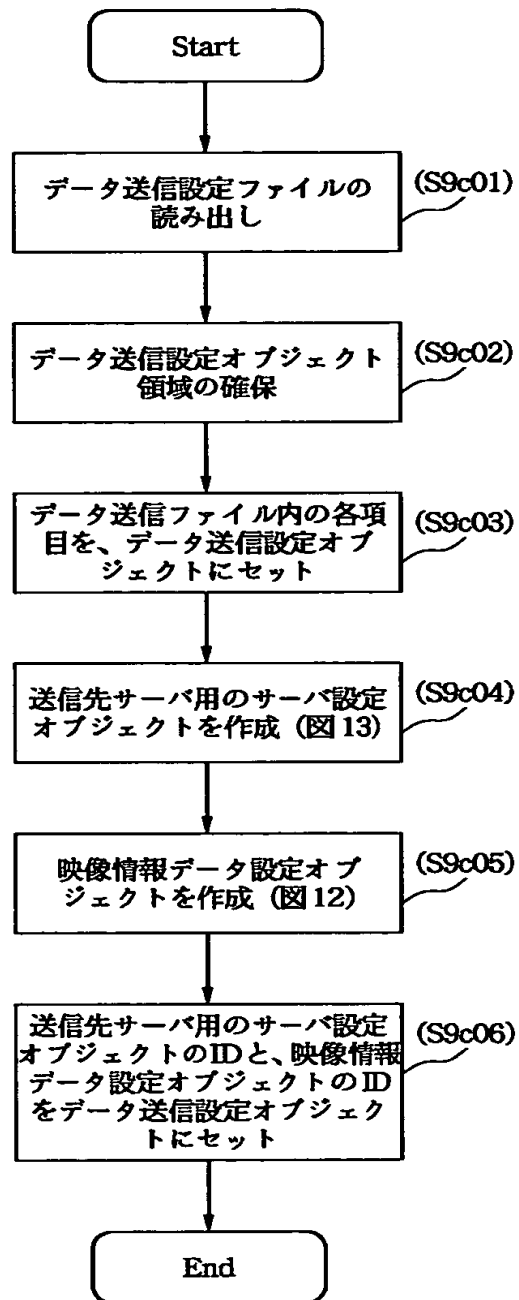
タイマ設定オブジェクトの作成

【図 1 0】



アップロード設定オブジェクトの作成

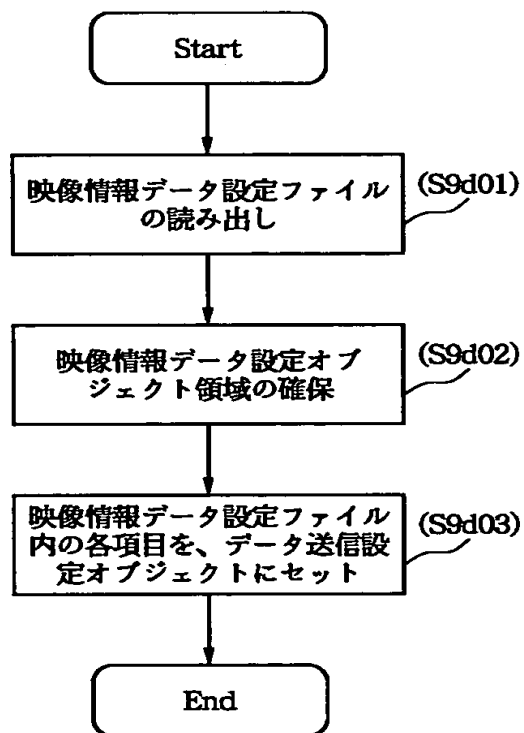
【図 1 1】



データ送信設定オブジェクトの作成

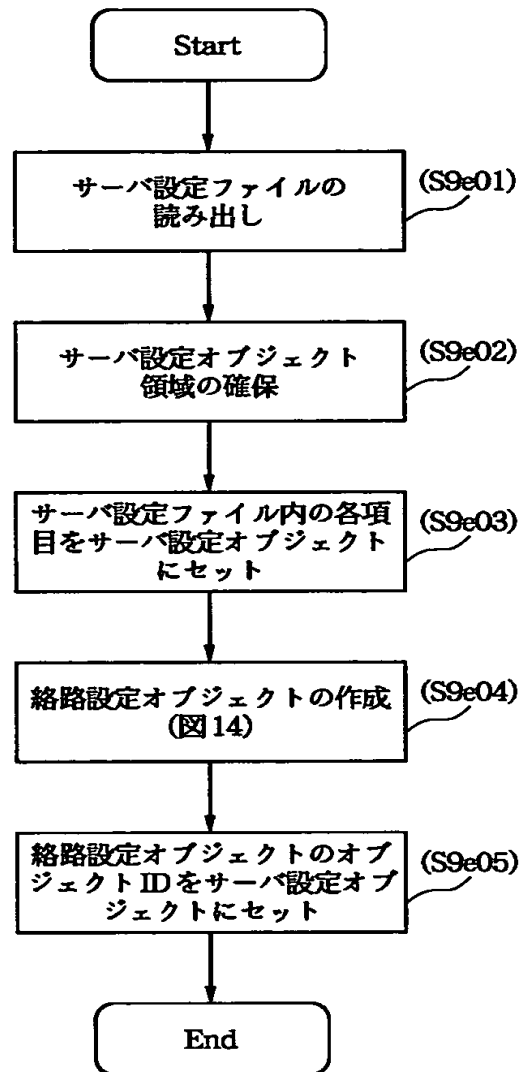


【図 1 2】



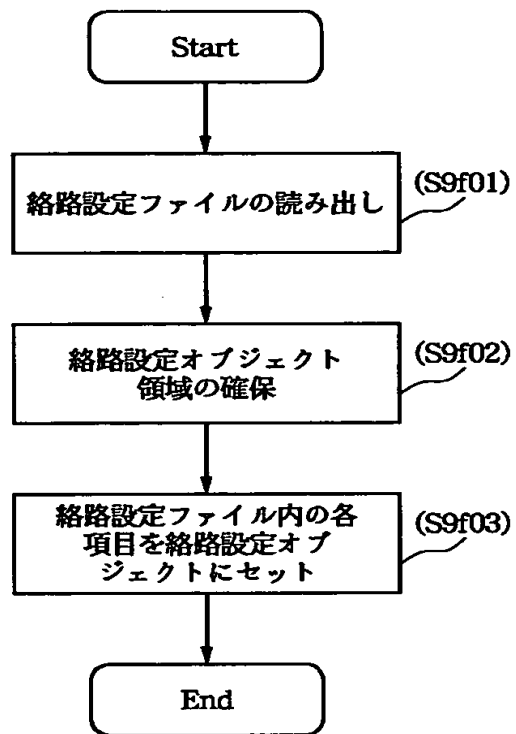
映像情報データ設定オブジェクトの作成

【図 1 3】



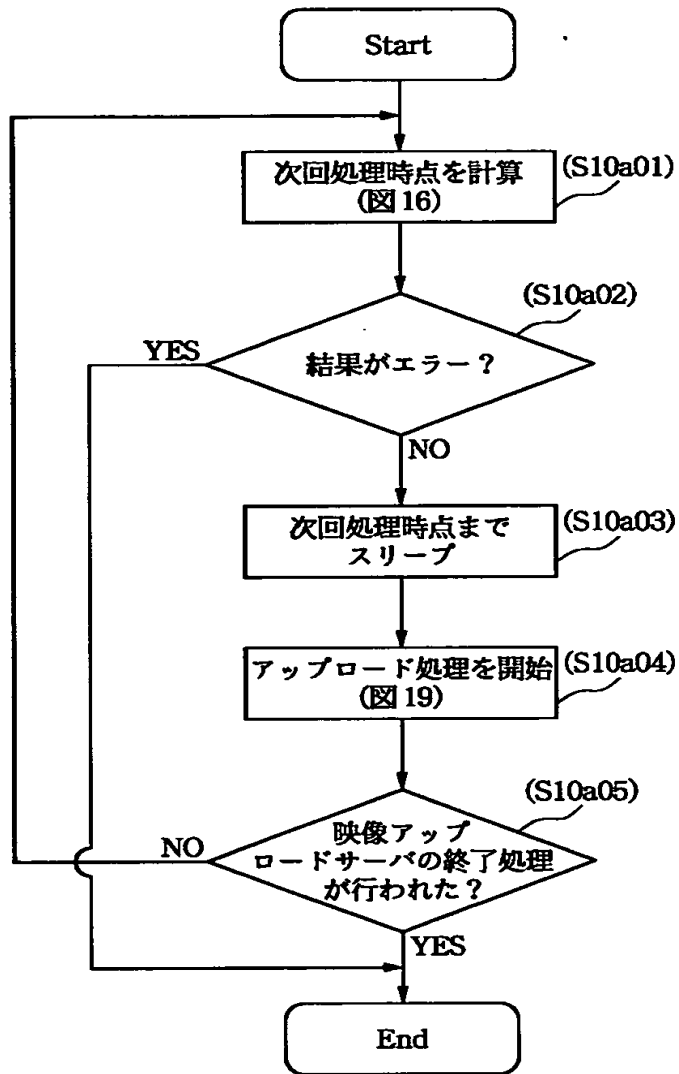
送信先サーバ用、又は、カメラサーバ用の  
サーバ設定オブジェクトの作成

【図 1 4】



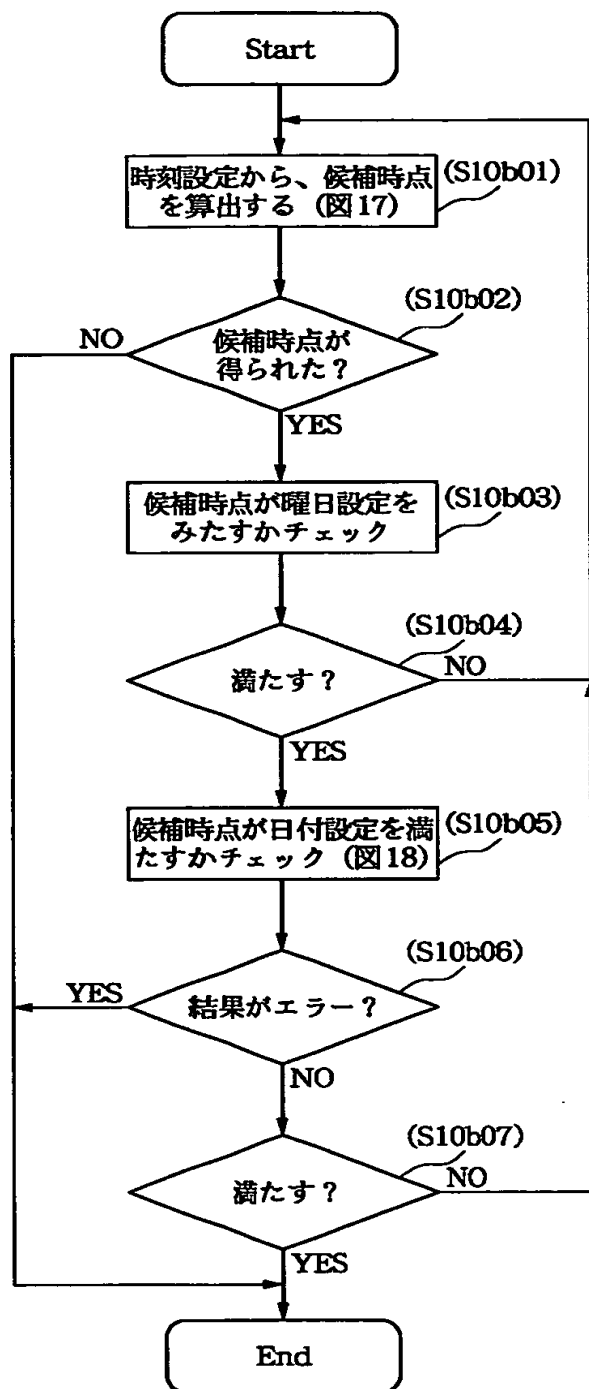
経路設定オブジェクトの作成

【図 1 5】



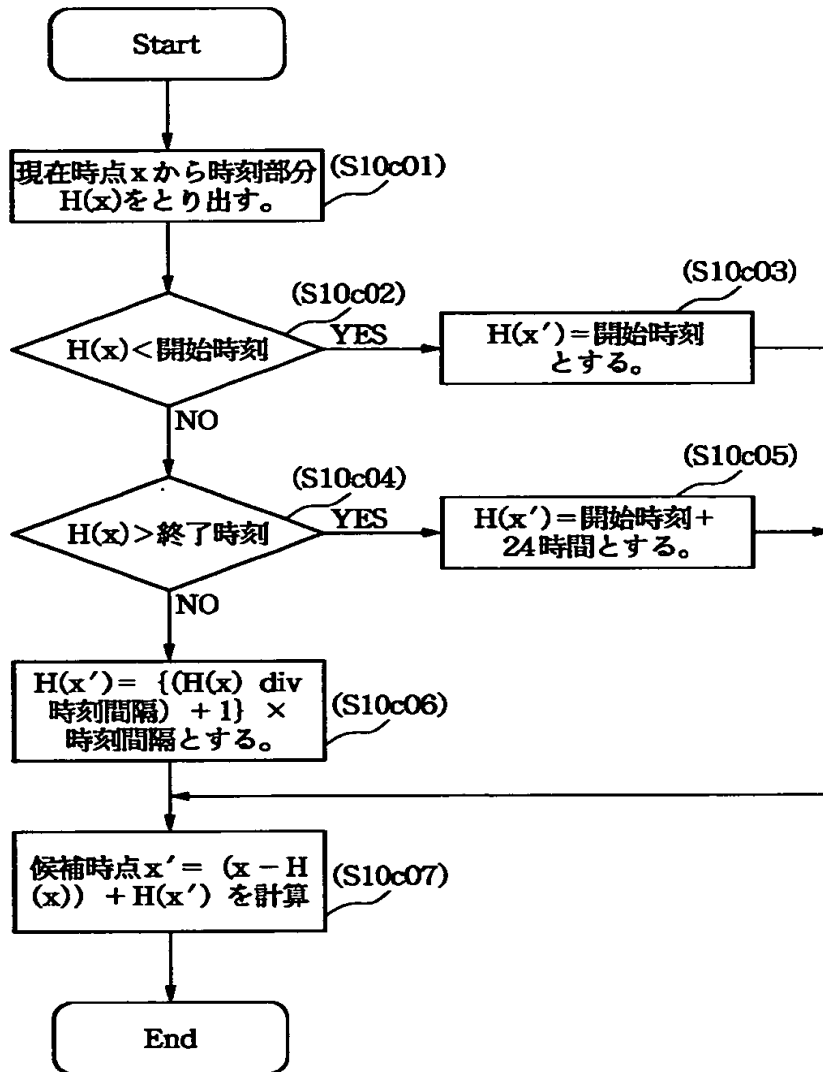
タイマ処理

【図 1 6】



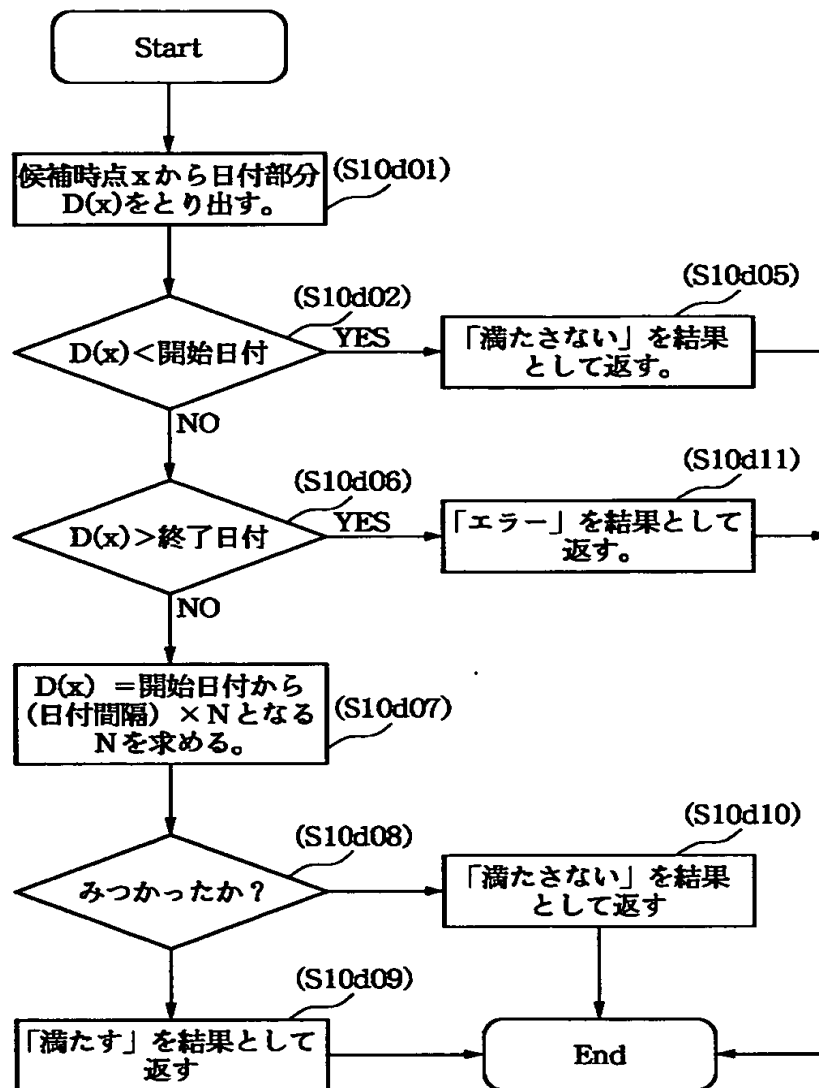
次回処理時点の計算

【図 1 7】



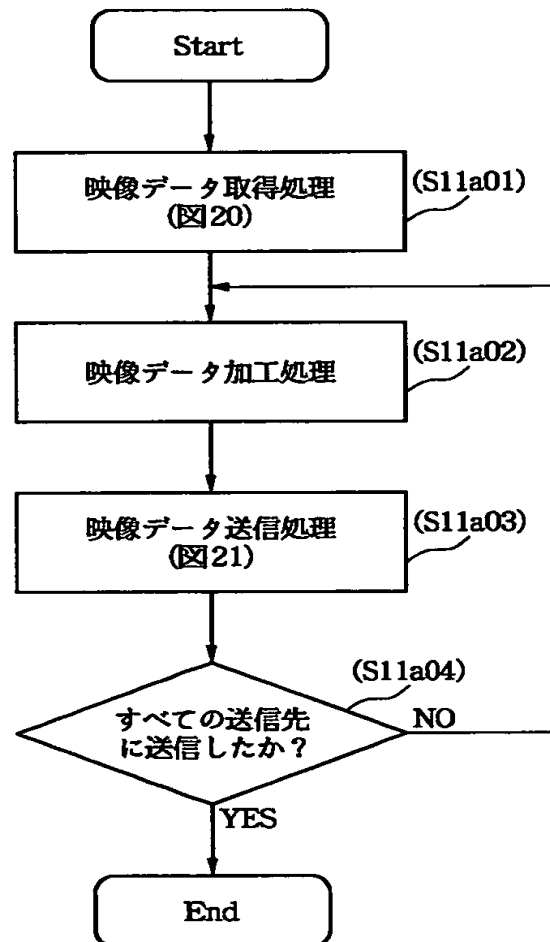
候補時点の算出

【図 1 8】



候補時点の日付設定のチェック

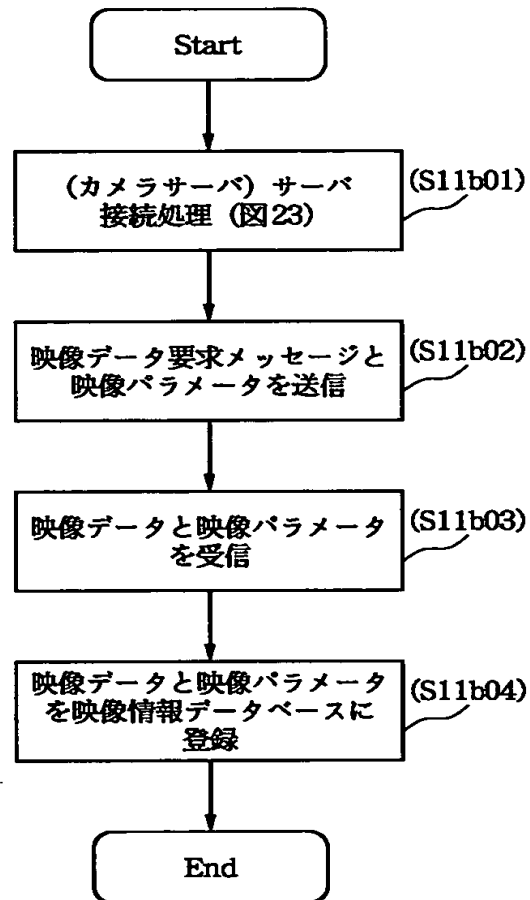
【図 1 9】



アップロード処理

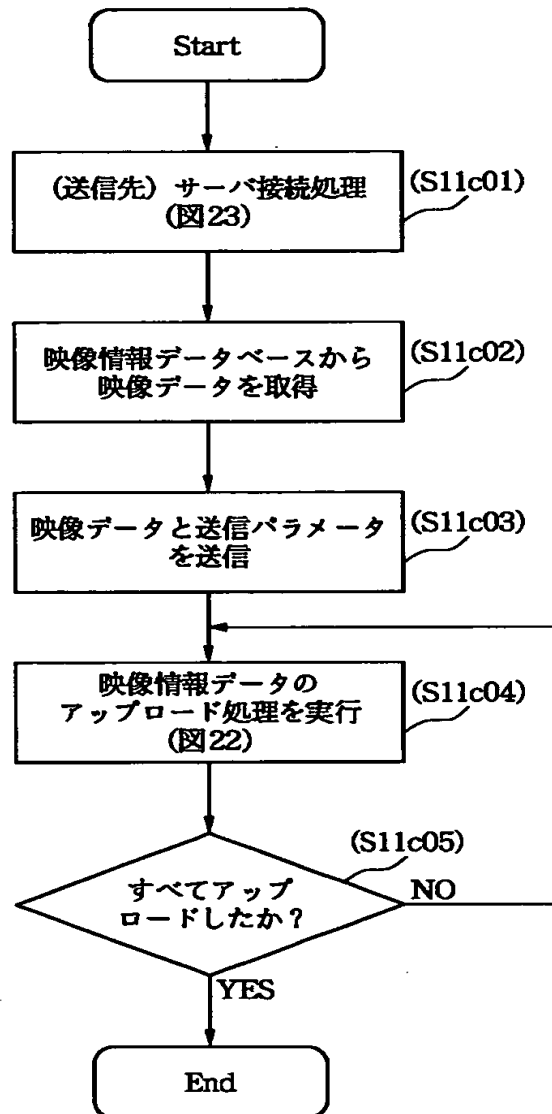


【図 2 0】



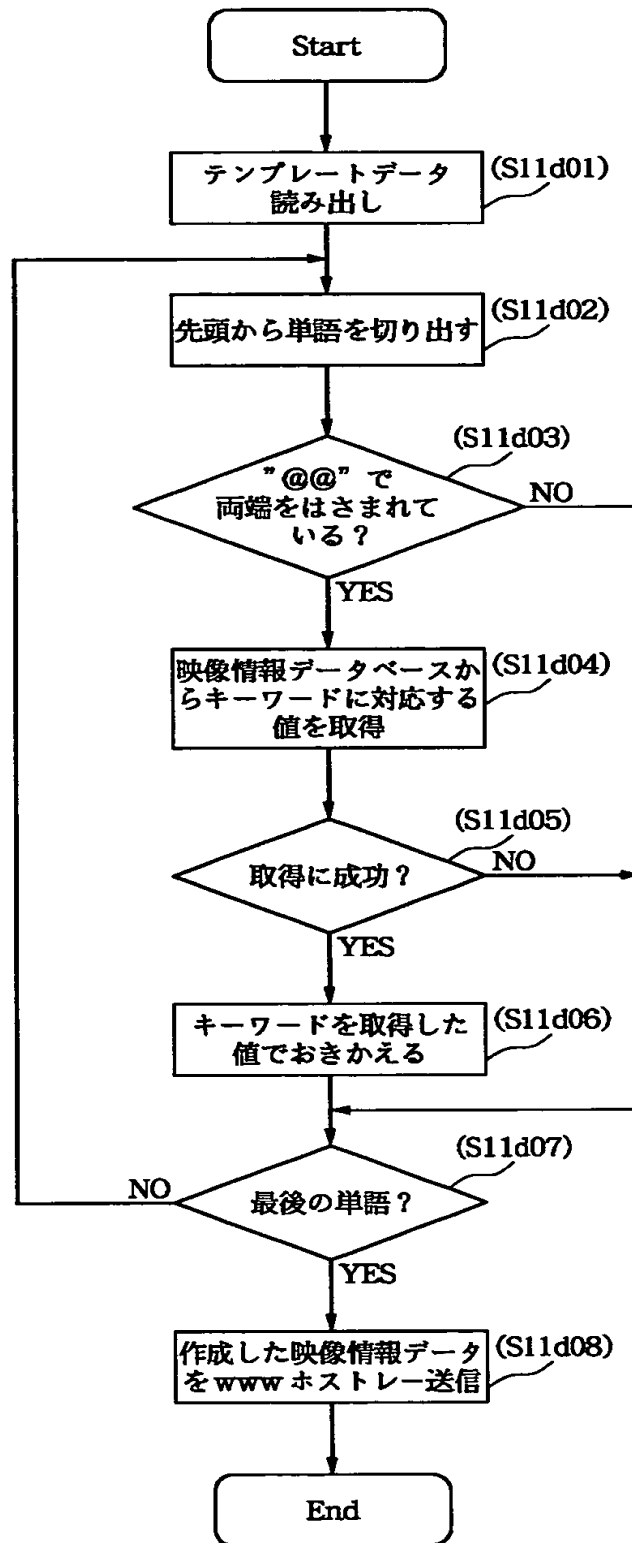
映像データ取得処理

【図 2 1】



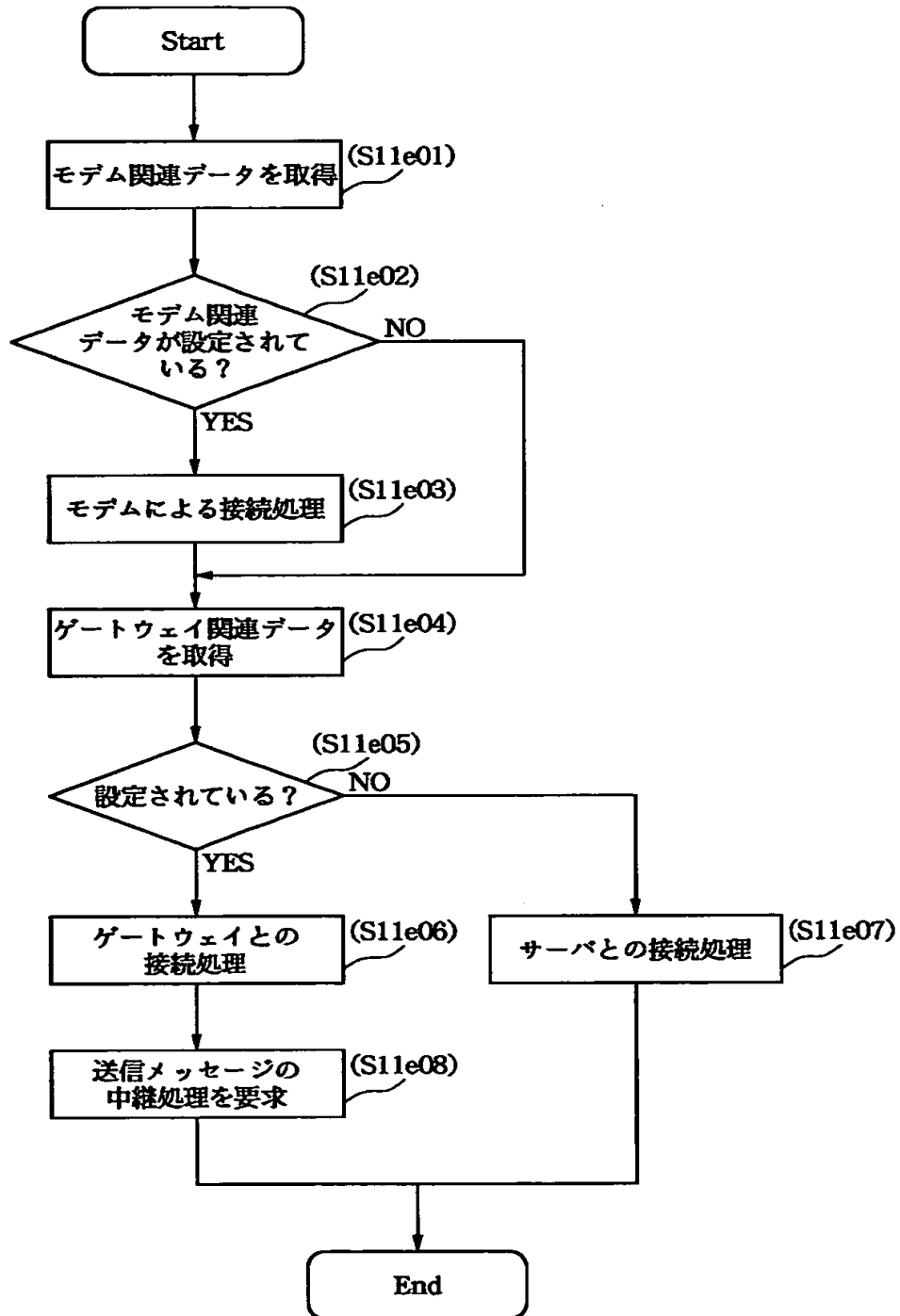
映像データ送信処理

【図 22】



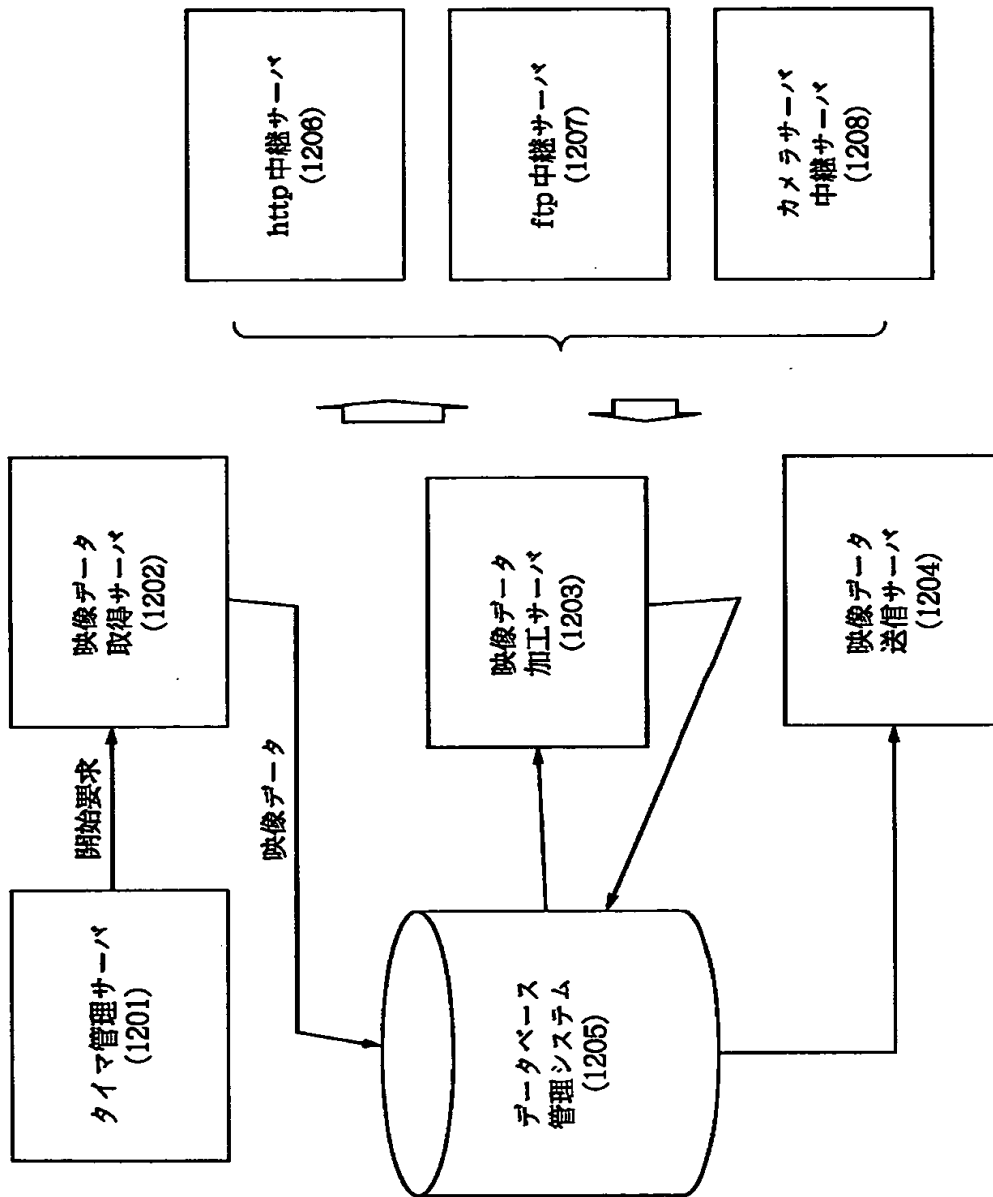
映像データのアップロード処理

【図 2 3】



サーバ接続処理

【 図 2 4 】



実施例 2 の構成図

【書類名】 要約書

【要約】

【課題】 映像データがWWWホスト(403)に記録されてからでないとな値がわからない情報もある。例えば映像データの取得時刻や映像データのサイズなどがこのような情報である。これらの情報をあらかじめ用意したHTMLファイル内に埋めこむことはできない。そこで、このような情報を表示するためには505のように、映像データの一部としてうめこまなければならなかった。

【解決手段】 本発明の映像アップロードシステムは、ネットワーク上に配置された映像入力装置から、映像データを取得する取得手段と、取得手段によって得られた複数の映像データを格納する格納手段と、予め指定されたスケジュールに従って、取得手段を制御する映像取得制御手段と、取得手段により得られた映像データに、当該映像取得時の情報を付加して、表示用映像を作成する作成手段と、作成手段によって作成された表示用映像を送信する送信手段とを有することを特徴とする。

【選択図】 図 7

出 願 人 履 歴 情 報

識別番号 [ 0 0 0 0 0 1 0 0 7 ]

1. 変更年月日 1 9 9 0 年 8 月 3 0 日

[ 変更理由 ] 新規登録

住 所 東京都大田区下丸子 3 丁目 3 0 番 2 号

氏 名 キヤノン株式会社